13

Algebraic Model for Agent Explicit Knowledge in Multi-agent Systems*

Khair Eddin Sabri, Ridha Khedri and Jason Jaskolka
Department of Computing and Software, McMaster University
Canada

1. Introduction

Information security is an important aspect that should be considered during system development. Analyzing the specification of a system enables detecting flaws at early stage of a system development. An agent knowledge of the exchanged information and its nature is essential for analyzing systems. An agent can enrich its knowledge by receiving information as messages and producing new information from the existing one. We classify an agent knowledge as *explicit knowledge* and *procedural knowledge*.
The *explicit knowledge* of an agent is related to the information that it possesses. For example in the context of a hospital software, an agent explicit knowledge would contain information about patients, drugs, and diseases. In the context of a school, the explicit knowledge of an agent would contain information about students, courses, and instructors. In the area of cryptographic protocols, the information of an agent can be its own key, the cipher used for encryption and decryption, and the identity of other agents that are involved in the protocol. Agents communicate by sending messages which are pieces of information stored in their explicit knowledge. The information an agent receives from other agents becomes a part of its explicit knowledge.
The *procedural knowledge* involves a set of mechanisms/operators that enables an agent to obtain new information from its explicit knowledge. For example, if the explicit knowledge of an agent contains an encrypted message as well as the key and the cipher used to decrypt the message, then by using the procedural knowledge, the concealed information can be obtained. The use of the procedural knowledge to analyze cryptographic protocols can be found in Sabri and Khedri (2006; 2007b).
The explicit knowledge representation is needed to analyze security related policies in multi-agent systems. We summarize below some uses of the explicit knowledge:

1. Agents communicate by exchanging messages, which are constructed from their explicit knowledge. Therefore, an agent explicit knowledge is necessary for modeling agents communications.
2. Explicit knowledge is required to specify agent internal actions such as verifying the existence of an information in the knowledge. The explicit knowledge representation becomes more useful in complex systems. For example, in the registration part of the *Equicrypt protocol* presented in Leduc and Germeau (2000), a third party can handle

*This chapter presents a revised and enlarged version of the material presented in Sabri et al. (2008)

simultaneously several registrations. Therefore, it should maintain an internal "table" with information on the users that have a registration in progress.

3. Some security properties are based on the explicit knowledge of agents. For example, a confidentiality security property would require that an agent should not know a specific kind of information existing in the explicit knowledge of another agent.
4. Even if the specification of a multi-agent system is proved to be secure by satisfying some security properties, it could contain flaws due to its incorrect implementation. To reduce the risk of incorrect implementation, one can derive the code automatically from the mathematical model of the system and prove that the derivation is correct. Having an explicit knowledge representation that allows specifying internal actions such as inserting and extracting information from the knowledge as well as verifying the existence of an information in the knowledge would be necessary for code generation.

For an efficient analysis of security policies in a multi-agent system, an explicit knowledge representation would have the following characteristics as giving in Sabri and Khedri (2008):

1. Classifying information so that one can reason on the ability of an agent to obtain an information that has a specific classification (e.g., private) in another agent's knowledge.
2. Relating information together such as relating patient to drugs so that one can reason on the ability of an agent to link pieces of information together.
3. Specifying internal actions such as inserting information into the knowledge and updating information.
4. Flexibility in specification by not having the same classification of information in all agents knowledge.
5. Specifying the explicit knowledge of systems with the same mathematical theory so that there is no need to introduce a new theory for a specific case.

In the literature, we find that explicit knowledge specifications satisfy some of the characteristics but not all of them. In this chapter, we present a mathematical structure to represent the explicit knowledge of agents that satisfies all the characteristics above. Then, we show that the structure is an *information algebra* which is introduced in Kohlas and Stärk (2007). In Section 2, we summarize information algebra. In Section 3, we present the mathematical structure to specify agent explicit knowledge. In Section 4, we give two applications of the uses of the proposed structure. In Section 5, we conclude.

2. Information Algebra

In Kohlas and Stärk (2007), the authors explore connections between different representations of information. They introduce a mathematical structure called *information algebra*. This mathematical structure involves a set of information Φ and a lattice D. They show that relational databases, modules, and constraint systems are information algebras. In the rest of this chapter, we denote elements of Φ by small letters of the Greek alphabet such as φ, ψ and χ. Each piece of information is associated with a *frame* (also called domain in Kohlas and Stärk (2007)), and the lattice D is the set of all frames. Each frame x contains a *unit element* e_x which represents the empty information. Information can be combined or restricted to a specific frame. Combining two pieces of information φ and ψ is represented by $\varphi\psi$. Information φ and ψ can be associated with different frames, and $\varphi\psi$ is associated with a more precise frame than φ and ψ. Kohlas and Stärk (2007) assume that the order of combining information does not matter

and, therefore, the combining operator is both commutative and associative. Restricting an information φ to a frame x is denoted by $\varphi^{\downarrow x}$ which represents only the part of φ associated with x.
In the following definition and beyond, let $(D, \curlyvee, \curlywedge)$ be a lattice and x and y be elements of D called frames. Let $\preccurlyeq$ be a binary relation between frames such that $x \curlyvee y = y \leftrightarrow x \preccurlyeq y$. Let Φ be a set of information and φ,ψ, χ be elements of Φ. We denote the frame of information $\varphi \in \Phi$ by $d(\varphi)$. Let e_x be the empty information over the frame $x \in D$, the operation $\downarrow$ be a partial mapping $\Phi \times D \to \Phi$, and $\cdot$ be a binary operator on information. For simplicity, to denote $\varphi \cdot \psi$, we write $\varphi\psi$.

Definition 1 (Information Algebra as in Kohlas and Stärk (2007))**.** *An information algebra is a system (Φ, D) that satisfies the following axioms:*

1. $(\varphi\psi)\chi = \varphi(\psi\chi)$
2. $\varphi\psi = \psi\varphi$
3. $d(\varphi\psi) = d(\varphi) \curlyvee d(\psi)$
4. $x \preccurlyeq y \to (e_y)^{\downarrow x} = e_x$
5. $d(\varphi) = x \to \varphi e_x = \varphi$
6. $\forall(x \mid x \in D : d(e_x) = x)$
7. $x \preccurlyeq d(\varphi) \to d(\varphi^{\downarrow x}) = x$
8. $x \preccurlyeq y \preccurlyeq d(\varphi) \to (\varphi^{\downarrow y})^{\downarrow x} = \varphi^{\downarrow x}$
9. $d(\varphi) = x \ \wedge \ d(\psi) = y \to (\varphi\psi)^{\downarrow x} = \varphi(\psi^{\downarrow x \curlywedge y})$
10. $x \preccurlyeq d(\varphi) \to \varphi\varphi^{\downarrow x} = \varphi$

□

The first two axioms indicate that the set of pieces of information together with the combining operator form a semi-group. Axiom 3 states that the frame of two pieces of information combined is the join of their frames. Axioms (4-6) give properties of the empty information e_x. Axioms (7-8) give the properties of focusing an information to a specific frame. Axioms (9-10) give properties that involve combining and focusing of information.

3. Specification of Agent Explicit Knowledge

In Sabri et al. (2008), we develop a mathematical structure to specify an agent explicit knowledge and prove that it is an information algebra. The explicit knowledge of an agent is represented by two elements Φ and D. The set Φ consists of pieces of information (we use the words information and piece of information interchangeably) available to the considered agent. There is no restriction on the representation of these pieces of information. They can be represented as formulae as in artificial intelligence literature, functions, etc. In this chapter, we represent pieces of information as functions. While D is a lattice of frames such that each piece of information is associated with a frame.

Definition 2 (Agent Information Frame)**.** *Let $\{\mathbb{A}_i \mid i \in I\}$ be a family of sets indexed by the set of indices I and $\mathcal{P}(\mathbb{A}_i)$ be the powerset of $\mathbb{A}_i$. An information frame D_I is defined as:*

$$D_I \triangleq \prod_{i \in I} \mathcal{P}(\mathbb{A}_i)$$

Which can be equivalently written as a set of functions as

$$D_I \triangleq \{f : I \to \bigcup_{i \in I} \mathcal{P}(\mathbb{A}_i) \mid \forall(i \mid i \in I : f(i) \in \mathcal{P}(\mathbb{A}_i))\}$$

Let $J \subseteq I$ and $\mathcal{I}_J \subseteq I \times I$ such that $\mathcal{I}_J = \{(x,x) \mid x \in J\}$ (i.e., $\mathcal{I}_J$ is the identity on J). Given the frame D_I, we can define D_J as $\{g \mid \exists(f \mid f \in D_I : g = \mathcal{I}_J ; f)\}$ where $;$ denotes relational composition. We call an element φ of D_J an *information* and D_J the *frame* of φ and denote[1] it by $d(\varphi)$. We call "d" the labelling operator. The information φ is a function which can be written as a set of 2-tuples (i,A) where i is an index and A is a set. Each frame D_J contains a special element called the *empty information* e_{D_J} and defined as $\{(i,\emptyset) \mid i \in J\}$. Whenever, it is clear from the context, we write e_J instead of e_{D_J}. We denote the *set of all frames* D_J for $J \subseteq I$ by D and the *set of all pieces of information* by Φ.

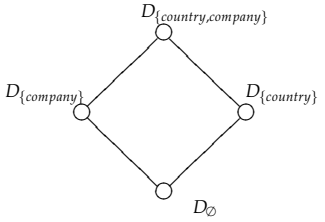

Fig. 1. A lattice constructed from $I = \{company, country\}$

As an example of our representation of Φ and D, suppose that an agent can handle only two kinds of information: *company* and *country*. In this case, the set of indices is $I = \{company, country\}$ and the lattice D is constructed as in Figure 1. The lattice D consists of four frames: $D_\emptyset$ is a frame that might involve only the empty information $e_\emptyset$ (absence of information), $D_{\{\text{company}\}}$ is the frame of the pieces of information classified as company, $D_{\{country\}}$ is the frame of the pieces of information classified as country, and $D_{\{company,\ country\}}$ is the frame of composite information where part of it is classified as company and another part is classified as country. Our aim from this lattice representation is to represent frames of atomic information as in $D_{\{country\}}$ and $D_{\{company\}}$ and to represent frames of composite information as in $D_{\{country,\ company\}}$.
To illustrate our representation of information, let the set of information Φ contains two pieces of information φ and ψ such that $\varphi = \{(company, \{AirFrance\}), (country, \{France\})\}$ and $\psi = \{(company, \{AirCanada\})\}$. The first information associates the company AirFrance with the country France while the second information contains the AirCanada information.

Definition 3. *An information φ is called atomic if $\varphi = e_\emptyset$ or $d(\varphi) = D_{\{j\}}$ for $j \in I$.*

From the definition, we can see that φ is a composite information while ψ is an atomic information. The set of information Φ can be represented in a tabular format as shown in Table 1. A piece of information can be seen as a row in a table where the table header represents the indices of the frame of an information. An empty information can be perceived as a table with only a header and $e_\emptyset$ can be seen as an empty page that does not contain even the header. An atomic information can be seen as a cell of the table or as an empty page. The following axiom and proposition are taken from Sabri et al. (2008) and are needed for the subsequent proofs.

[1]The notation $d(\varphi)$ to denote the frame of φ comes from the usage of the term domain in Kohlas and Stärk (2007) as a synonym for frame. We prefer to use the term frame to avoid any confusion with the domain of a relation.

Table 1. The set Φ in a tabular format

	company	*country*
φ	AirFrance	France
ψ	AirCanada	-

Axioms 1. 1. $\varphi \in D_J \rightarrow d(\varphi) = D_J$ 2. $e_J \triangleq \{(i,\emptyset) \mid i \in J\}$ □

From the definition of D_J, it follows that $\varphi \in D_J \rightarrow \forall(i \mid i \in J : \varphi(i) \in \mathcal{P}(\mathbb{A}_i))$. Therefore, $\varphi \in D_J$ can be written as a set of 2-tuples $\{(i,A) \mid i \in J \wedge A \subseteq \mathbb{A}_i\}$.

Proposition 1. *For $J, K \subseteq I$ and $\varphi \in \Phi$, we have:*

1. $\varphi \in D_K \rightarrow \mathcal{I}_J{}^{\cdot}\varphi = \{(i,A) \mid i \in (J \cap K) \wedge A \subseteq \mathbb{A}_i\}$
2. $\mathcal{I}_J{}^{\cdot}\mathcal{I}_K = \mathcal{I}_{J\cap K}$
3. $\varphi \in D_K \rightarrow d(\mathcal{I}_J{}^{\cdot}\varphi) = D_{J\cap K}$
4. $\mathcal{I}_{J\cup K} = \mathcal{I}_J \cup \mathcal{I}_K$

Proof.
1. The proof invokes the definitions of relation composition, φ, and $\mathcal{I}_J$ as well as the trading rule for $\exists$, set intersection axiom, and the distributivity of $\wedge$ over $\exists$.
2. The proof invokes the definition of D_K, definition of $D_{J\cap K}$, Proposition 1(2), and Axiom 1(1).
3. One uses the definition of $\mathcal{I}_J$, $\mathcal{I}_K$, and $\mathcal{I}_{J\cup K}$ as well as applies set union axiom and range split axiom.

The complete proof is given in Sabri and Khedri (2007a). □

We define a binary operator $\cdot$ to combine information (we write $\varphi\psi$ to denote $\varphi \cdot \psi$). We can use this operator to represent composite information made of pieces of information.

Definition 4 (Combining Information)**.** *Let Φ be a set of information and φ, ψ be its elements. Let $d(\varphi) = D_J$ and $d(\psi) = D_K$. We define the binary operator $\cdot$ (however, we write $\varphi\psi$ to denote $\varphi \cdot \psi$) on information as: $\varphi\psi \triangleq \{(i,A) \mid i \in J \cap K \wedge A = \varphi(i) \cup \psi(i)\} \cup \{(i,A) \mid i \in J - K \wedge A = \varphi(i)\} \cup \{(i,A) \mid i = K - J \wedge A = \psi(i)\}$* □

We also define two operators on frames as follows:

Definition 5. *Let D_J and D_K be frames and $\varphi * \psi = \{(i,A) \mid i \in J \cap K \wedge A = \varphi(i) \cap \psi(i)\}$, we define the operators $\curlyvee$ and $\curlywedge$ on frames as:*

1. $D_J \curlyvee D_K \triangleq \{\chi \mid \exists(\varphi,\psi \mid \varphi \in D_J \wedge \psi \in D_K : \chi = \varphi\psi)\}$
2. $D_J \curlywedge D_K \triangleq \{\chi \mid \exists(\varphi,\psi \mid \varphi \in D_J \wedge \psi \in D_K : \chi = \varphi * \psi)\}$ □

Proposition 2. $D_J \curlyvee D_K = D_{J\cup K}$

Proof. The proof calls for the definitions of D_J, D_K, and $D_{J\cup K}$ as well as Definition 5(1), distributivity of $\wedge$ over $\exists$, trading rule for $\exists$, nesting axiom, interchange of dummies, Definition 4, Proposition 1(1), renaming, and range split axiom. The detailed proof is given in Appendix A. □

Proposition 3. $D_J \curlywedge D_K = D_{J \cap K}$

Proof. The proof is similar to that of Proposition 2. We use the definitions of D_J, D_K, and $D_{J \cap K}$ and we apply Definition 5(2), distributivity of $\wedge$ over $\exists$, trading rule for $\exists$, nesting axiom, interchange of dummies, Proposition 1(1), renaming, and range split axiom. The complete proof is given in Sabri and Khedri (2007a). □

Proposition 4. *Let D_J, D_K, and D_K be frames, we have*

1. $D_J \curlyvee D_K = D_K \curlyvee D_J$
2. $D_J \curlywedge D_K = D_K \curlywedge D_J$
3. $(D_J \curlyvee D_K) \curlyvee D_L = D_J \curlyvee (D_K \curlyvee D_L)$
4. $(D_J \curlywedge D_K) \curlywedge D_L = D_J \curlywedge (D_K \curlywedge D_L)$
5. $D_J \curlyvee (D_J \curlywedge D_L) = D_J$
6. $D_J \curlywedge (D_J \curlyvee D_L) = D_J$

Proof. We use Proposition 2, Proposition 3 and the properties of $\cap$ and $\cup$. The complete proof is given in Sabri and Khedri (2007a). □

The following proposition is a consequence result of Proposition 4.

Proposition 5 (Lattice of Frames)**.** $(\{D_J\}_{J \subseteq I}, \curlyvee, \curlywedge)$ *form a lattice.*

For simplicity, we use D to denote the lattice $(\{D_J\}_{J \subseteq I}, \curlyvee, \curlywedge)$. On the lattice D and for D_J and D_K frames in D, it is known that the following are equivalent (Davey and Priestley, 2002, page 39):

1. $D_J \preccurlyeq D_K$
2. $D_J \curlyvee D_K = D_K$
3. $D_J \curlywedge D_K = D_J$

We define a partial order relation $\leq$ on information as $\varphi \leq \psi$ and we say that ψ is *more informative* than φ.

Definition 6 (More Informative Relation)**.** *Let Φ be a set of information and φ, ψ be elements of Φ. Let D be a lattice and D_J and D_K be elements of D. Let $d(\varphi) = D_J$ and $d(\psi) = D_K$. We define the binary relation $\leq$ on information as: $\varphi \leq \psi \leftrightarrow J \subseteq K \ \wedge \ \forall (i \mid i \in J : \varphi(i) \subseteq \psi(i))$* □

The relation $\leq$ indicates whether or not an information is a part of another one. We use it to verify the existence of an information in the knowledge of an agent. An information can be in the knowledge of an agent as a part of a composite information. The special element $e_\emptyset$ of $D_\emptyset$ is the least informative information i.e., $\forall (\varphi \mid \varphi \in \Phi : e_\emptyset \leq \varphi)$.

Proposition 6. *The relation $\leq$ is a partial order.*

Proof. The proof is based on the property that $\subseteq$ is a partial order. The proof is given in Sabri and Khedri (2007a). □

We show in Sabri et al. (2008) that there is a relation between frames and their indices.

Proposition 7.

1. $\forall(J,K \mid J,K \subseteq I : J = K \rightarrow D_J = D_K)$
2. $\forall(J,K \mid J,K \subseteq I : D_J = D_K \rightarrow J = K)$
3. $\forall(J,K \mid J,K \subseteq I : D_J \preccurlyeq D_K \leftrightarrow J \subseteq K)$

Proof. 1. The proof uses trading rule for $\forall$, Substitution axiom, and properties of propositional logic.

2. We prove by contrapositive. We assume that $J \neq K$ and prove $D_J = D_K \rightarrow \mathsf{false}$. The proof uses the definition of D_J and D_K, definition of "$\leftrightarrow$", Weakening, Proposition 1(4), the distributivity of relational composition over $\cup$, Distributivity of $\wedge$ over $\exists$, $\exists$-True body, and properties of propositional logic.
3. The proof uses Proposition 7(2), Proposition 2, Reflexivity of $\leftrightarrow$, $\forall$-True body, and properties of set theory.

The complete proof is given in Appendix A. □

We also define a binary operator to extract a part of an information that belongs to a specific frame as:

Definition 7 (Marginalizing Information)**.** *Let D_J be a frame and φ be an information such that $D_J \in D$ and $\varphi \in \Phi$, we define a binary operator $\downarrow : \Phi \times D \rightarrow \Phi$ as $\varphi^{\downarrow D_J} \triangleq \mathcal{I}_J{:}\varphi$.* □

The $\downarrow$ operator can be used to extract a specific kind of information. For example, let $\varphi = \{(company, \{AirFrance\}), (country, \{France\})\}$, then $\varphi^{\downarrow D\{company\}} = \{(company, \{AirFrance)\}$. After defining information marginalizing, labelling and combination in our context, we prove in Sabri et al. (2008) that our structure is an information algebra by proving the following proposition.

Proposition 8. *For $J,K \subseteq I$, we have*

1. $(\varphi\psi)\chi = \varphi(\psi\chi)$
2. $\varphi\psi = \psi\varphi$
3. $d(\varphi\psi) = d(\varphi) \curlyvee d(\psi)$
4. $d(\varphi) = D_J \rightarrow \varphi e_J = \varphi$
5. $d(e_J) = D_J$
6. $D_J \preccurlyeq D_K \rightarrow (e_K)^{\downarrow D_J} = e_J$
7. $d(\varphi) = D_J \wedge d(\psi) = D_K \rightarrow (\varphi\psi)^{\downarrow D_J} = \varphi(\psi^{\downarrow D_J \wedge D_K})$
8. $D_J \preccurlyeq d(\varphi) \rightarrow d(\varphi^{\downarrow D_J}) = D_J$
9. $D_J \preccurlyeq D_K \preccurlyeq d(\varphi) \rightarrow (\varphi^{\downarrow D_K})^{\downarrow D_J} = \varphi^{\downarrow D_J}$
10. $D_J \preccurlyeq d(\varphi) \rightarrow \varphi\varphi^{\downarrow D_J} = \varphi$

Proof. 1. The proof calls for Definition 4, commutativity and associativity of $\cup$, and properties of set difference.

2. We use Definition 4 and commutativity of $\cap$ and $\cup$.
3. The proof essentially invokes Axiom 1(1), Propositions 2, the definition of $D_{J\cup K}$, Proposition 1(1), and Definition 4.
4. We basically use Definition 4, Axiom 1(2), idempotency of $\cap$, and empty range axiom.

5. The proof essentially calls for Axiom 1(1, 2), the definition of D_J, and Proposition 1(1).
6. The proof invokes Definition 7, Axiom 1(2), Proposition 1(1), and Proposition 7(3).
7. The proof invokes Definition 7, Definition 4, Proposition 1(1), and properties of set difference, $\cup$ and $\cap$.
8. The proof invokes Definition 7, Proposition 1(3), Axiom 1(1), and Proposition 7(3).
9. We use Definition 7, Proposition 1(2), and Proposition 7(3).
10. The proof calls for Definition 7, Proposition 1(1), Definition 4, Axiom 1(1), Proposition 7(3), range split axiom, and properties of set difference, $\cup$, and $\cap$.

The full detailed proof can be found in Sabri and Khedri (2007a). □

Proposition 9. *The structure (Φ, D) is an information algebra.*

Proof. (Φ, D) satisfies the ten axioms of information algebra (see Definition 1) as shown in Proposition 8. □

As consequence results of proving that (Φ, D) is an information algebra, the following properties hold and the proofs can be found in Kohlas and Stärk (2007):

Proposition 10.

1. $d(\varphi) = D_J \rightarrow \varphi^{\downarrow D_J} = \varphi$
2. $\varphi\varphi = \varphi$
3. $D_J \preccurlyeq D_K \rightarrow e_J e_K = e_K$
4. $d(\varphi) = D_J \rightarrow (\varphi D_K)^{\downarrow D_J} = \varphi$
5. $D_J \preccurlyeq d(\varphi) \rightarrow (\varphi e_K)^{\downarrow D_J} = \varphi^{\downarrow D_J}$
6. $e_{J \curlyvee K} = e_J e_K$

□

The empty information has some interesting properties as shown in the following proposition.

Proposition 11. *Let $\varphi \in D_J$ and $J \subseteq I$, we have*

1. $\varphi^{\downarrow D_\emptyset} = e_\emptyset$
2. $e_\emptyset^{\downarrow D_J} = e_\emptyset$
3. $\varphi e_\emptyset = \varphi$
4. $e_\emptyset = \emptyset$

Proof. The proof uses Definition 7, Definition 4, Proposition 1(1), Axiom 1, and properties of set theory and propositional logic. The full proof is given in Sabri et al. (2009a). □

We also prove some properties related to the marginalzing operator.

Proposition 12. *Let $\varphi \in D_L$ and $\psi \in D_K$, we have*

1. $\varphi^{\downarrow D_J} \cdot \varphi^{\downarrow D_K} = \varphi^{\downarrow D_{J \cup K}}$
2. $(\varphi\psi)^{\downarrow D_L} = \varphi^{\downarrow D_L} \psi^{\downarrow D_L}$
3. $\varphi^{\downarrow D_K} = \varphi^{\downarrow D_{J \cap K}}$
4. $(\varphi^{\downarrow D_K})^{\downarrow D_L} = \varphi^{\downarrow D_{K \cap L}}$

Proof. The proof uses the definition of $\cdot$, the definition of $\downarrow$, Proposition 1(1), and properties of set theory and propositional logic. The full proof is given in Appendix A. □

In addition to the information algebra operators, we define in Sabri et al. (2009b) an operator to remove a piece of information from another one.

Definition 8 (Removing Information)**.** *Let $d(\varphi) = D_J$ and $d(\psi) = D_K$. We define the binary operator "$-$" as:* $\varphi - \psi \triangleq \{(i, A) \mid i \in J \cap K \wedge A = \varphi(i) - \psi(i)\} \cup \{(i, A) \mid i \in J - K \wedge A = \varphi(i)\}$ □

Let $\varphi = \{(company, \{AirFrance\}), (country, \{France\})\}$ and $\psi = \{(company, \{AirFrance\})\}$, then $\varphi - \psi = \{(company, \{\}), (country, \{France\})\}$. We also prove in Sabri et al. (2009b) the following proposition.

Proposition 13. *Let φ, ψ and χ be pieces of information such that $d(\varphi) = D_J$, $d(\psi) = D_K$, and $d(\chi) = D_L$. Also, let e_K be the empty information on D_K*

1. $d(\varphi - \psi) = d(\varphi)$
2. $\varphi - e_K = \varphi$
3. $e_K - \varphi = e_K$
4. $\varphi \leq (\psi - \chi) \rightarrow \varphi \leq \psi$
5. $\varphi \leq \psi \rightarrow \varphi - \psi = e_{d(\varphi)}$
6. $(\varphi\psi - \psi)^{\downarrow_{d(\varphi)}} \leq \varphi$
7. $\varphi \leq \psi \rightarrow (\chi - \varphi)\psi = \chi\psi$

Proof.

1. The proof uses Axiom 1(1), Definition of D_J, Distributivity of $;$ over $\cup$, Proposition 1(1), Empty range axiom, Definition of $\varphi - \psi$, and properities of set theory and propositional logic.
2. The proof uses Definition of $\varphi - e_K$, Definition of e_K, Distributivity of $\wedge$ over $\vee$, and properities of set theory.
3. The proof uses Definition of $e_K - \varphi$, Definition of e_K, Distributivity of $\wedge$ over $\vee$, and properities of set theory.
4. The proof uses Definition of $\leq$, Range split for $\forall$, Empty range axiom, Definition of $\varphi - \chi$, Distributivity axiom, Weakening, and properties of set theory and propositional logic.
5. The proof uses Definition of $\varphi - \psi$, Empty range, Definition of e_J, and properties of set theory and propositional logic.
6. The proof uses Definition of $\varphi\psi - \psi$, Definition of $\downarrow_{D_J}$, Distributivity of $;$ over $\cup$, Proposition 1(1), Definition of combining information, Definition of $\leq$, $\forall$-True body, and properties of set theory and propositional logic.
7. The proof uses Definition of $\leq$, The definition of combining information, Range split, Definition of set difference, Definition of $\chi - \varphi$, and properties of set theory and propositional logic.

Full proof is given in Sabri et al. (2009a). □

The proposition gives some properties of the remove operator such as Proposition 13(1) which indicates that removing pieces from an information does not change the frame of that information. Proposition 13(2, 3) states that removing an empty piece from an information does not affect that information, and a removing piece of information from the empty information does not change the empty information. Also, the proposition relates the more informative relation with the remove operator as shown in Proposition 13(4,5). Proposition 13(6,7) relates the remove operator with the combine operator.

We note that agents might have different lattices of frames. The frame of an information at a sender's knowledge might be assigned to a different frame after its transmission to a receiver. In Sabri et al. (2009b), we define a frame substitution function that substitutes a part of a frame of an information with another as:

Definition 9 (Frame Substitution). *$fs(\varphi, D_J, D_K) \triangleq \varphi^{\downarrow D_{(L-J)}} \cdot (\varphi^{\downarrow D_J}[D_K/D_J])$ where the sets J and K are singleton subsets of the set of indices I and $d(\varphi) = D_L$.* □

We note that this function is defined using basic information algebra operators. As an example of the frame substitution function, let $\varphi = \{(country,\{France\}),(company,\{AirFrance\})\}$. Then, $fs(\varphi, D_{\{company\}}, D_{\{airline\}}) = \{(country,\{France\}),(airline,\{AirFrance\})\}$.
We prove a proposition related to set theory that we used in Sabri et al. (2009b) to prove properties related to frame substitution function.

Proposition 14. *For $J = \{j\}$, we have $\neg(J \subseteq K) \rightarrow J \cap K = \emptyset$*

Proof. The proof uses properties of set theory. The full proof is given in Sabri et al. (2009a). □

Proposition 15. *Let $J = \{j\}$ and $K = \{k\}$ be singleton subsets of the set of indices I and $d(\varphi) = D_L$, we have*

1. $D_J \preccurlyeq d(\varphi) \quad \vee \quad \varphi = fs(\varphi, D_J, D_K)$
2. $D_K \preccurlyeq d(\varphi)) \ \vee \ \varphi = fs(fs(\varphi, D_J, D_K), D_K, D_J)$

Proof.
1. The proof uses Proposition 7(3), Proposition 14, Definition of $fs(\varphi, D_J, D_K)$, Proposition 10(1), Definition of $\downarrow_{D_J}$, Proposition 1(1), Proposition 11(4), Proposition 11(3), and properties of set theory and propositional logic.
2. To prove $\varphi = fs(fs(\varphi, D_J, D_K), D_K, D_J)$, we have two cases: $\neg(D_J \preccurlyeq d(\varphi))$ and $D_J \preccurlyeq d(\varphi)$. The proof of the first case uses Proposition 7(3), Proposition 14, and Proposition 15(1). The proof of the second case uses Definition of *fs*, Proposition 1(2, 9), Proposition 11(1), Proposition 11(3), Proposition 12(2), Proposition 12(4), Proposition 12(3), and Proposition 12(1).

The full proof is given in Appendix A. □

As discussed in Sabri et al. (2008; 2009b), the *knowledge* of each agent is modeled as an information algebra $\mathcal{N} \triangleq (\Phi, D)$. Based on the operators of information algebra, we introduce in Sabri et al. (2008; 2009b) several functions to specify operations on knowledge.

- $isInKnowledge(\mathcal{N}, x, \varphi) \triangleq \exists(\psi \mid \psi \in \Phi : x \in D \ \wedge \ \varphi \leq \psi \ \wedge \ x \preccurlyeq d(\psi))$. This function verifies the existence of an information in the knowledge $\mathcal{N}$ associated with the frame x and is more informative than φ.
- $extract(\mathcal{N}, x, \varphi) \triangleq \{\psi^{\downarrow x} \mid x \in D \ \wedge \ \psi \in \Phi \ \wedge \ \varphi \leq \psi \ \wedge \ x \preccurlyeq d(\psi)\}$. This function extracts pieces of information from the knowledge $\mathcal{N}$ that contains φ and restricts them to the frame x.
- $insert(\mathcal{N}, \varphi)$. This function inserts the information φ into Φ.
- $update(\mathcal{N}, \psi, \varphi) \triangleq (\{(\chi - \psi) \cdot \varphi \mid \chi \in \Phi \ \wedge \ \psi \leq \chi\} \ \cup \ \{\chi \mid \chi \in \Phi \ \wedge \ \neg(\psi \leq \chi)\}, D)$. This function update the knowledge $\mathcal{N}$ by replacing ψ with φ.

In the insert and update functions, there is always a condition that $d(\varphi) \in D$. We also define in Sabri et al. (2009b) the function $choose(\Phi)$ to select a piece of information randomly from Φ. If Φ is empty, it returns the empty information $e_\emptyset$. In Sabri et al. (2009b), we prove the following proposition which helps in verifying policies.

Proposition 16. *Let φ and ψ be pieces of information and let $\mathcal{N}$ be a knowledge.*

1. $\varphi \leq \psi \ \wedge \ \varphi \leq \chi \rightarrow update(update(\mathcal{N}, \varphi, \psi), \varphi, \chi) = update(\mathcal{N}, \varphi, \psi \cdot \chi)$
2. $isInKnowledge(\mathcal{N}, d(\varphi), \varphi) \ \vee \ update(\mathcal{N}, \varphi, \psi) = \mathcal{N}$

Proof. 1. The proof uses the definition of the function *update*, Distributivity axiom, Trading rule for $\exists$, Nesting axiom, Distributivity of $\wedge$ over $\vee$, Proposition 13(7), Substitution axiom, and properties of set theory and propositional logic.

2. The proof uses Definition of *isInKnowledge*, De Morgan laws, Proposition 7(1), Definition of *update*, Empty range axiom, and properties of set theory and propositional logic.

The full proof is given in Appendix A. □

4. Application

The proposed mathematical structure has several applications in the analysis of security properties. We summarize here its use in the analysis of cryptographic protocols and information flow. We implement a prototype tool in the functional programming language *Haskell*. This prototype tool is used to represent and manipulate explicit knowledge of agents. It allows initializing the lattice of frames D and the set of information Φ for each agent. It implements the functions presented earlier so that the user can insert, remove and update the knowledge of each agent. Also, it allows extracting information from the knowledge and verifying the existence of an information in the knowledge of an agent.

4.1 Cryptographic Protocols

In Sabri et al. (2008), we show the use of our representation of the explicit knowledge and its functions to specify protocols, specify properties, reduce the state space, and generate a specific type of attack with the aid of the developed prototype tool.

- Specify protocol: the tool allows specifying the insertion of information and the update of the knowledge

```
-- insertInformation is the implementation of the
-- function insert presented in the previous section.

-- insertInformation function inserts the key "hello"
-- into the frame named "key" at the knowledge of agent "S".

insertInformation "S"
   ([("key",["hello"])],["key"])
```

- Specify properties: the tool allows specifying several properties such as an intruder Z should not get a session key *"hello"*.

```
-- isInKnowledge is the implementation of the
-- function isInKnowledge presented in the previous section.

-- isInKnowledge function checks if the intruder knowledge (Z)
-- contains the key "hello" associated with the frame "key"

isInKnowledge "Z" (["key"])
   ([("key",["k"])],["key"])
```

- Reduce state space: the tool allows specifying intruder that send useful messages. For example, all the messages sent to the server should be encrypted with the server public key if the server should decrypt the message.

```
-- extractInformation is the implementation of the
-- function extract presented in the previous section.

-- extractInformation extracts from the knowledge Z the public keys that
-- are associated with the server.

extractInformation "Z" (["publicK"])
   ([("id",["Server"])],["id"])
```

- Generate attack: the tool allows mounting a specific kind of attack such as a reflection attack where the intruder Z sends messages back to the sender.

```
-- extractInformation extract from the knowledge Z all the messages
-- that are associated with the sender John Do.

extractInformation "Z" (["message"])
   ([("sender",["John Do"])],["sender"])
```

In Sabri et al. (2008), we summarize existing techniques used to specify agent knowledge in cryptographic protocols. For instance, Leduc and Germeau (2000) adopt LOTOS, Fábrega et al. (1999) propose strand space, Clarke et al. (2000) introduce Brutus, Paulson (1998) adopt an inductive approach, Ma and Cheng (2005) introduced a knowledge-based logical system, and finally Cervesato (2000) introduce MSR. We also compare them to our mathematical structure. We show that our explicit knowledge representation allows specifying knowledges similar to the existing techniques. However, our mathematical structure allows the specifier to define only a set of frames of the explicit knowledge which indicates the classification of information. There is no need to specify the relation between information as in the existing techniques such as relating public key with a private key. We use a compact number of operators to specify the agent explicit knowledge of any protocol. For example, the knowledge-based logical approach as found in Ma and Cheng (2005) uses about six functions to specify the registration phase of the SET protocol. Four functions are used to map an agent to its public encryption key, private encryption key, public signature key, and private signature keys. Also, a function is introduced to associate two agents with a shared key, and another function to verify if a message is a part of another one. In our structure, only the pre-defined operators within the framework are required to manipulate the information. There is no need to define new operators. Having a small number of operators would reduce the complexity of specifying cryptographic protocols and verifying them.
Also, the proposed framework enables specifying the internal actions of agents. For example, we can specify the ability of the server to check the freshness of a message while this is not possible in Brutus as we find in Clarke et al. (2000). The inability of specifying the internal actions would affect the protocol analysis and implementation.

4.2 Information Flow Analysis

In Sabri et al. (2009b), we apply our explicit knowledge structure in developing a technique to verify information flow in agent-based systems. The technique is based on information algebra to represent agent knowledge, global calculus to represent the communication and an amended version of Hoare logic for verification. We use Hoare triple $\{P\}S\{Q\}$ to conduct verification where the precondition P represents a condition on the initial knowledge of agents, S represents the specification of the communication between agents, and the postcondition Q represents the negation of a confidentiality policy on the knowledge of agents. The

precondition and the postcondition are expressed within the language of information algebra. To verify a policy, we first calculate the weakest precondition from S and Q and then prove or disprove that $P \rightarrow wp(S,Q)$. The inference rules are obtained by amending Hoare's set of rules to make them appropriate to protocols specified using global calculus and information algebra. For more details, we refer the reader to Sabri et al. (2009b). A tool is used in Sabri et al. (2009b) together with the PVS theorem prover to verify policies.
In Sabri et al. (2009b), we show that the use of information algebra to specify confidentiality policies allows specifying policies similar to that of Bell and LaPadula (1976) and Brewer and Nash (1989) models. Also, it allows analyzing composite information flow, which is not taken into consideration in the existing techniques such as Alghathbar et al. (2006); Focardi and Gorrieri (1997); Hristova et al. (2006); Varadharajan (1990). Analyzing composite information enables verifying the possibility of an agent to link pieces of information together and therefore, build up an important composite information.

5. Conclusion

In this chapter, we present a structure to specify agent explicit knowledge based on information algebra. We define in the context of agent knowledge the combining, marginalizing, and labelling operators. Also, we define remove and frame substitution operator. These operators are all what is needed to express operations on agent explicit knowledge. We also define a set of frames to be associated with information. Then, we prove that our structure is an information algebra which links our work to a rich heritage of mathematical theories. Our mathematical structure is expressive as it allows combining information for different purposes regardless of their frames, extracting a part of information, or associating information with a frame.
We give two applications of the proposed structure. First, we apply it to the specification and analysis of agent knowledge in cryptographic protocols. In the literature of cryptographic protocols, operators are usually defined on information that belongs to a specific type, while our structure enables a uniform and a general way to handle information. Also, defining a relation between frames and linking them to the operators applied on information is not addressed in the literature. Furthermore, different protocol-dependent structures should be defined to relate different kinds of information which are not needed in our representation. Second, we show its use in the analysis of information flow between agents in multi-agent systems. Our structure provides a comprehensive language to specify agents knowledge and confidentiality policies. For example, it allows specifying and reasoning on composite information flow. Also, it allows specifying policies similar those articulated within Bell-LaPadula and Chinese Wall models.

A. Detailed Proofs

A.1 Proposition 2

$D_J \curlyvee D_K = D_{J \cup K}$

Proof.

$$
\begin{aligned}
& D_J \curlyvee D_K \\
= & \qquad \langle \text{ Definition 5(1) } \rangle \\
& \{\chi \mid \exists(\varphi, \psi \mid \varphi \in D_J \wedge \psi \in D_K : \chi = \varphi\psi)\}
\end{aligned}
$$

$$
\begin{array}{ll}
= & \langle\ y \in \{x \mid r\} \leftrightarrow r[x := y]\ \text{\& Definition of } D_J\ \rangle \\
& \{\chi \mid \exists(\varphi, \psi \mid \exists(f \mid f \in D_I : \varphi = \mathcal{I}_J;f) \wedge \psi \in D_K : \chi = \varphi\psi)\} \\
= & \langle\ y \in \{x \mid r\} \leftrightarrow r[x := y]\ \text{\& Definition of } D_K\ \rangle \\
& \{\chi \mid \exists(\varphi, \psi \mid \exists(f \mid f \in D_I : \varphi = \mathcal{I}_J;f) \\
& \wedge\ \exists(g \mid g \in D_I : \psi = \mathcal{I}_K;g) : \chi = \varphi\psi)\} \\
= & \langle\ \text{Distributivity of } \wedge \text{ over } \exists\ \rangle \\
& \{\chi \mid \exists(\varphi, \psi \mid \exists(f \mid f \in D_I : \varphi = \mathcal{I}_J;f \\
& \wedge\ \exists(g \mid g \in D_I : \psi = \mathcal{I}_K;g)) : \chi = \varphi\psi)\} \\
= & \langle\ \text{Trading rule for } \exists\ \rangle \\
& \{\chi \mid \exists(\varphi, \psi \mid \exists(f \mid f \in D_I \wedge \varphi = \mathcal{I}_J;f : \\
& \exists(g \mid g \in D_I : \psi = \mathcal{I}_K;g)) : \chi = \varphi\psi)\} \\
= & \langle\ \text{Nesting axiom}\ \rangle \\
& \{\chi \mid \exists(\varphi, \psi \mid \\
& \exists(f, g \mid f \in D_I \wedge \varphi = \mathcal{I}_J;f \wedge g \in D_I : \psi = \mathcal{I}_K;g) : \chi = \varphi\psi)\} \\
= & \langle\ \text{Trading rule for } \exists \text{ \& Symmetry of } \wedge\ \rangle \\
& \{\chi \mid \exists(\varphi, \psi \mid \\
& \exists(f, g \mid f \in D_I \wedge g \in D_I : \varphi = \mathcal{I}_J;f \wedge \psi = \mathcal{I}_K;g) : \chi = \varphi\psi)\} \\
= & \langle\ \text{Trading rule for } \exists\ \rangle \\
& \{\chi \mid \exists(\varphi, \psi \mid : \\
& \exists(f, g \mid f \in D_I \wedge g \in D_I : \varphi = \mathcal{I}_J;f \wedge \psi = \mathcal{I}_K;g) \wedge \chi = \varphi\psi)\} \\
= & \langle\ \text{Distributivity of } \wedge \text{ over } \exists\ \rangle \\
& \{\chi \mid \exists(\varphi, \psi \mid : \\
& \exists(f, g \mid f \in D_I \wedge g \in D_I : \varphi = \mathcal{I}_J;f \wedge \psi = \mathcal{I}_K;g \wedge \chi = \varphi\psi))\} \\
= & \langle\ \text{Substitution axiom}\ \rangle \\
& \{\chi \mid \exists(\varphi, \psi \mid : \\
& \exists(f, g \mid f \in D_I \wedge g \in D_I : \varphi = \mathcal{I}_J;f \wedge \psi = \mathcal{I}_K;g \wedge \chi = (\mathcal{I}_J;f)\psi))\} \\
= & \langle\ \text{Substitution axiom}\ \rangle \\
& \{\chi \mid \exists(\varphi, \psi \mid : \\
& \exists(f, g \mid f \in D_I \wedge g \in D_I : \varphi = \mathcal{I}_J;f \wedge \psi = \mathcal{I}_K;g \wedge \chi = (\mathcal{I}_J;f)(\mathcal{I}_K;g)))\} \\
= & \langle\ \text{Trading rule for } \exists \text{ \& Symmetry of } \wedge\ \rangle \\
& \{\chi \mid \exists(\varphi, \psi \mid : \\
& \exists(f, g \mid f \in D_I \wedge g \in D_I \wedge \chi = (\mathcal{I}_J;f)(\mathcal{I}_K;g) : \varphi = \mathcal{I}_J;f \wedge \psi = \mathcal{I}_K;g))\} \\
= & \langle\ \text{Interchange of dummies}\ \rangle \\
& \{\chi \mid \exists(f, g \mid f \in D_I \wedge g \in D_I \wedge \chi = (\mathcal{I}_J;f)(\mathcal{I}_K;g) : \\
& \exists(\varphi, \psi \mid : \varphi = \mathcal{I}_J;f \wedge \psi = \mathcal{I}_K;g))\} \\
= & \langle\ \mathcal{I}_x \text{ and the functions } f \text{ and } g \text{ are always defined, and there composition is} \\
& \quad \text{defined as well}\ \rangle \\
& \{\chi \mid \exists(f, g \mid f \in D_I \wedge g \in D_I \wedge \chi = (\mathcal{I}_J;f)(\mathcal{I}_K;g) : \mathsf{true})\} \\
= & \langle\ \text{Trading rule for } \exists \text{ \& Identity of } \wedge\ \rangle \\
& \{\chi \mid \exists(f, g \mid f \in D_I \wedge g \in D_I : \chi = (\mathcal{I}_J;f)(\mathcal{I}_K;g))\}
\end{array}
$$

$$
\begin{array}{ll}
= & \langle \text{ Definition 4 and Proposition 1(1) } \rangle \\
& \{\chi \mid \exists(f,g \mid f,g \in D_I : \chi = \{(i,A) \mid i \in J \cap K \ \wedge \ A = f(i) \cup g(i)\} \\
& \cup\{(i,A) \mid i \in J - K \ \wedge \ A = f(i)\} \\
& \cup\{(i,A) \mid i \in K - J \ \wedge \ A = g(i)\})\} \\
= & \langle \text{ Let } h(i) = \begin{cases} f(i) \cup g(i) & \text{if } i \in (J \cap K), \\ f(i) & \text{if } i \in (J - K), \\ g(i) & \text{if } i \in (K - J). \\ \emptyset & \text{if } i \in I - (J \cup K) \end{cases} \\
& \quad \rangle \\
& \{\chi \mid \exists(h \mid h \in D_I : \chi = \{(i,A) \mid i \in J \cap K \ \wedge \ A = h(i)\} \\
& \cup\{(i,A) \mid i \in J - K \ \wedge \ A = h(i)\} \\
& \cup\{(i,A) \mid i \in K - J \ \wedge \ A = h(i)\})\} \\
= & \langle \text{ Range split axiom } \rangle \\
& \{\chi \mid \exists(h \mid h \in D_I : \chi = \{(i,A) \mid i \in (J \cap K) \cup (J - K) \cup (K - J) \ \wedge \ A = h(i)\})\} \\
= & \langle \text{ Set theory } \rangle \\
& \{\chi \mid \exists(h \mid h \in D_I : \chi = \{(i,A) \mid i \in J \cup K \ \wedge \ A = h(i)\})\} \\
= & \langle \text{ Proposition 1(1) } \rangle \\
& \{\chi \mid \exists(h \mid h \in D_I : \chi = \mathcal{I}_{J \cup K}; h)\} \\
= & \langle \text{ Definition of } D_{J \cup K} \rangle \\
& D_{J \cup K}
\end{array}
$$

□

A.2 Proposition 7

1. $\forall(J,K \mid J,K \subseteq I : J = K \rightarrow D_J = D_K)$
2. $\forall(J,K \mid J,K \subseteq I : D_J = D_K \rightarrow J = K)$
3. $\forall(J,K \mid J,K \subseteq I : D_J \preccurlyeq D_K \leftrightarrow J \subseteq K)$

Proof. 1.

$$
\begin{array}{ll}
& \forall(J,K \mid J,K \subseteq I : J = K \rightarrow D_J = D_K) \\
\leftarrow & \langle \text{ Trading rule for } \forall \rangle \\
& \forall(J,K \mid J,K \subseteq I \ \wedge \ J = K : D_J = D_K) \\
\leftarrow & \langle \text{ Trading rule for } \forall \ \& \ p \ \wedge \ q \rightarrow p \rangle \\
& \forall(J,K \mid J,K \subseteq I : (J = K \ \wedge \ D_J = D_K) \leftrightarrow J = K) \\
\leftarrow & \langle \text{ Substitution axiom } \rangle \\
& \forall(J,K \mid J,K \subseteq I : (J = K \ \wedge \ D_K = D_K) \leftrightarrow J = K) \\
\leftarrow & \langle \ A = A \leftrightarrow \mathsf{true} \ \rangle \\
& \forall(J,K \mid J,K \subseteq I : (J = K \ \wedge \ \mathsf{true}) \leftrightarrow J = K) \\
\leftarrow & \langle \ (p \ \wedge \ \mathsf{true}) \leftrightarrow p \ \rangle \\
& \forall(J,K \mid J,K \subseteq I : J = K \leftrightarrow J = K) \\
\leftarrow & \langle \ (p \leftrightarrow p) \leftrightarrow \mathsf{true} \ \rangle \\
& \forall(J,K \mid J,K \subseteq I : \mathsf{true}) \\
\leftarrow & \langle \ \forall\text{-True body} \ \rangle
\end{array}
$$

true

2.

$$\begin{array}{ll} & \forall(J,K \mid J,K \subseteq I : D_J = D_K \rightarrow J = K) \\ \leftrightarrow & \quad \langle \text{ Contrapositive } \rangle \\ & \forall(J,K \mid J,K \subseteq I : J \neq K \rightarrow D_J \neq D_K) \end{array}$$

To prove the proposition we assume that $J \neq K$ and prove $D_J = D_K \rightarrow \mathsf{false}$ (which is equivalent to $\neg(D_J = D_K)$).

$$\begin{array}{ll} & D_J = D_K \\ \leftrightarrow & \quad \langle\ y \in \{x \mid r\} \leftrightarrow r[x := y] \text{ \& Definition of } D_J \text{ and } D_K\ \rangle \\ & \{g \mid \exists(f \mid f \in D_I : g = \mathcal{I}_J; f)\} = \{g \mid \exists(f \mid f \in D_I : g = \mathcal{I}_K; f)\} \\ \leftrightarrow & \quad \langle\ \{x \mid Q\} = \{x \mid R\} \leftrightarrow \forall(x \mid\!: Q \leftrightarrow R)\ \rangle \\ & \forall(g \mid\!: \exists(f \mid f \in D_I : g = \mathcal{I}_J; f) \leftrightarrow \exists(f \mid f \in D_I : g = \mathcal{I}_K; f)) \\ \rightarrow & \quad \langle \text{ Definition of "}\leftrightarrow\text{" \& Weakening } \rangle \\ & \forall(g \mid\!: \exists(f \mid f \in D_I : g = \mathcal{I}_J; f) \rightarrow \exists(f \mid f \in D_I : g = \mathcal{I}_K; f)) \\ \rightarrow & \quad \langle\ J \neq K \text{ \& Assume } K = J \cup \{k\}\ \rangle \\ & \forall(g \mid\!: \exists(f \mid f \in D_I : g = \mathcal{I}_J; f) \rightarrow \exists(f \mid f \in D_I : g = \mathcal{I}_{J\cup\{k\}}; f)) \\ \rightarrow & \quad \langle \text{ Proposition 1(4) \& Relational composition distributes over } \cup\ \rangle \\ & \forall(g \mid\!: \exists(f \mid f \in D_I : g = \mathcal{I}_J; f) \rightarrow \exists(f \mid f \in D_I : g = \mathcal{I}_J; f \cup \mathcal{I}_{\{k\}}; f)) \\ \rightarrow & \quad \langle\ J \subseteq \mathsf{dom}\,(f)\ \rangle \\ & \forall(g \mid\!: \exists(f \mid f \in D_I : |g| = |J|) \rightarrow \exists(f \mid f \in D_I : |g| = |J| + 1)) \\ \rightarrow & \quad \langle \text{ Distributivity of } \wedge \text{ over } \exists\ \rangle \\ & \forall(g \mid\!: \exists(f \mid f \in D_I : \mathsf{true}) \wedge |g| = |J| \rightarrow \exists(f \mid f \in D_I : \mathsf{true}) \wedge |g| = |J| + 1) \\ \rightarrow & \quad \langle\ \exists\text{-True body \& Identity of } \wedge\ \rangle \\ & \forall(g \mid\!: |g| = |J| \rightarrow |g| = |J| + 1) \\ \rightarrow & \quad \langle \text{ Implication (i.e., } (p \rightarrow q) \leftrightarrow (\neg p \vee q))\ \rangle \\ & \forall(g \mid\!: |g| \neq |J| \vee |g| = |J| + 1) \\ \rightarrow & \quad \langle \text{ Let } g = e_J \text{ \& } |g| = |J| \text{ \& } \mathsf{false} \vee \mathsf{false} \rightarrow \mathsf{false}\ \rangle \\ & \mathsf{false} \end{array}$$

3.

$$\begin{array}{ll} & \forall(J,K \mid J,K \subseteq I : D_J \preccurlyeq D_K \leftrightarrow J \subseteq K) \\ \leftarrow & \quad \langle\ J \cup K = K \leftrightarrow J \subseteq K\ \rangle \\ & \forall(J,K \mid J,K \subseteq I : D_J \preccurlyeq D_K \leftrightarrow J \cup K = K) \\ \leftarrow & \quad \langle \text{ Proposition 7(2) } \rangle \\ & \forall(J,K \mid J,K \subseteq I : D_J \preccurlyeq D_K \leftrightarrow D_{J\cup K} = D_K) \\ \leftarrow & \quad \langle \text{ Proposition 2 } \rangle \\ & \forall(J,K \mid J,K \subseteq I : D_J \preccurlyeq D_K \leftrightarrow D_J \curlyvee D_K = D_K) \\ \leftarrow & \quad \langle\ D_J \preccurlyeq D_K \leftrightarrow D_J \curlyvee D_K = D_K\ \rangle \\ & \forall(J,K \mid J,K \subseteq I : D_J \curlyvee D_K = D_K \leftrightarrow D_J \curlyvee D_K = D_K) \\ \leftarrow & \quad \langle \text{ Reflexivity of } \leftrightarrow\ \rangle \\ & \forall(J,K \mid J,K \subseteq I : \mathsf{true}) \\ \leftarrow & \quad \langle\ \forall\text{-True body}\ \rangle \\ & \mathsf{true} \end{array}$$

□

A.3 Proposition 12

1. $\varphi^{\downarrow_{D_J}} \cdot \varphi^{\downarrow_{D_K}} = \varphi^{\downarrow_{D_{J\cup K}}}$
2. $(\varphi\psi)^{\downarrow_{D_L}} = \varphi^{\downarrow_{D_L}}\psi^{\downarrow_{D_L}}$
3. $\varphi^{\downarrow_{D_K}} = \varphi^{\downarrow_{D_{J\cap K}}}$
4. $(\varphi^{\downarrow_{D_K}})^{\downarrow_{D_L}} = \varphi^{\downarrow_{D_{K\cap L}}}$

Proof. 1.

$$
\begin{array}{ll}
 & \varphi^{\downarrow_{D_J}} \cdot \varphi^{\downarrow_{D_K}} \\
= & \qquad \langle \text{ Definition of } \downarrow_{D_J} \text{ and } \downarrow_{D_K} \rangle \\
 & \mathcal{I}_J{;}\varphi \cdot \mathcal{I}_K{;}\varphi \\
= & \qquad \langle\ \varphi \in D_L\ \rangle \\
 & \mathcal{I}_J{;}\{(i,A) \mid i \in L \wedge A = \varphi(i)\} \cdot \mathcal{I}_K{;}\{(i,A) \mid i \in L \wedge A = \varphi(i)\} \\
= & \qquad \langle \text{ Proposition 1(1) } \rangle \\
 & \{(i,A) \mid i \in L \cap J \wedge A = \varphi(i)\} \cdot \{(i,A) \mid i \in L \cap K \wedge A = \varphi(i)\} \\
= & \qquad \langle \text{ Definition of } \cdot \rangle \\
 & \quad \{(i,A) \mid i \in (L \cap J) \cap (L \cap K) \wedge A = \varphi(i) \cup \varphi(i)\} \\
 & \cup \{(i,A) \mid i \in (L \cap J) - (L \cap K) \wedge A = \varphi(i)\} \\
 & \cup \{(i,A) \mid i \in (L \cap K) - (L \cap J) \wedge A = \varphi(i)\} \\
= & \qquad \langle \cup \text{ is idempotent } \rangle \\
 & \quad \{(i,A) \mid i \in (L \cap J) \cap (L \cap K) \wedge A = \varphi(i)\} \\
 & \cup \{(i,A) \mid i \in (L \cap J) - (L \cap K) \wedge A = \varphi(i)\} \\
 & \cup \{(i,A) \mid i \in (L \cap K) - (L \cap J) \wedge A = \varphi(i)\} \\
= & \qquad \langle \text{ Range split (i.e., } \{x \mid r\} \cup \{x \mid p\} = \{x \mid r \vee p\}) \rangle \\
 & \{(i,A) \mid i \in (L \cap J) \cap (L \cap K) \wedge A = \varphi(i) \\
 & \ \vee\ i \in (L \cap J) - (L \cap K) \wedge A = \varphi(i) \\
 & \ \vee\ i \in (L \cap K) - (L \cap J) \wedge A = \varphi(i)\} \\
= & \qquad \langle \text{ Distributivity of } \wedge \text{ over } \vee \rangle \\
 & \{(i,A) \mid (i \in (L \cap J) \cap (L \cap K) \vee i \in (L \cap J) - (L \cap K) \vee i \in (L \cap K) - (L \cap J)) \\
 & \ \wedge\ A = \varphi(i)\} \\
= & \qquad \langle \text{ Set union axiom (i.e., } i \in A \vee i \in B \leftrightarrow i \in A \cup B) \rangle \\
 & \{(i,A) \mid i \in ((L \cap J) \cap (L \cap K)) \cup ((L \cap J) - (L \cap K)) \cup ((L \cap K) - (L \cap J)) \\
 & \ \wedge\ A = \varphi(i)\} \\
= & \qquad \langle \text{ Set theory } \rangle \\
 & \{(i,A) \mid i \in (L \cap (K \cup J)) \ \wedge\ A = \varphi(i)\} \\
= & \qquad \langle \text{ Proposition 1(1) } \rangle \\
 & \mathcal{I}_{J\cup K}{;}\{(i,A) \mid i \in L \ \wedge\ A = \varphi(i)\} \\
= & \qquad \langle\ \varphi \in D_L\ \rangle \\
 & \mathcal{I}_{J\cup K}{;}\varphi \\
= & \qquad \langle \text{ Definition of } \downarrow_{D_{J\cup K}} \rangle \\
 & \varphi^{D_{J\cup K}}
\end{array}
$$

2.

$$
\begin{array}{ll}
 & (\varphi\psi)^{\downarrow_{D_L}} \\
= & \qquad \langle \text{ Definition of } \cdot \rangle \\
 & \quad (\{(i,A) \mid i \in J \cap K \wedge A = \varphi(i) \cup \psi(i)\} \\
 & \cup \{(i,A) \mid i \in J - K \wedge A = \varphi(i)\} \\
 & \cup \{(i,A) \mid i \in K - J \wedge A = \psi(i)\})^{\downarrow_{D_L}} \\
= & \qquad \langle \text{ Definition of } \downarrow_{D_L} \rangle
\end{array}
$$

$$
\begin{array}{ll}
& \mathcal{I}_L;\{(i,A) \mid i \in J \cap K \ \wedge \ A = \varphi(i) \cup \psi(i)\} \\
& \cup \ \mathcal{I}_L;\{(i,A) \mid i \in J - K \ \wedge \ A = \varphi(i)\} \\
& \cup \ \mathcal{I}_L;\{(i,A) \mid i \in K - J \ \wedge \ A = \psi(i)\} \\
= & \qquad \langle \text{ Proposition 1(1) } \rangle \\
& \{(i,A) \mid i \in J \cap K \cap L \ \wedge \ A = \varphi(i) \cup \psi(i)\} \\
& \cup \ \{(i,A) \mid i \in (J - K) \cap L \ \wedge \ A = \varphi(i)\} \\
& \cup \ \{(i,A) \mid i \in (K - J) \cap L \ \wedge \ A = \psi(i)\} \\
= & \qquad \langle \text{ Set theory } \rangle \\
& \{(i,A) \mid i \in (J \cap L) \cap (K \cap L) \ \wedge \ A = \varphi(i) \cup \psi(i)\} \\
& \cup \ \{(i,A) \mid i \in (J \cap L) - (K \cap L) \ \wedge \ A = \varphi(i)\} \\
& \cup \ \{(i,A) \mid i \in (K \cap L) - (J \cap L) \ \wedge \ A = \psi(i)\} \\
= & \qquad \langle \text{ Definition of } \cdot \text{ and } \downarrow_{D_L} \rangle \\
& \varphi^{\downarrow_{D_L}} \psi^{\downarrow_{D_L}}
\end{array}
$$

3.

$$
\begin{array}{ll}
& \varphi^{\downarrow_{D_K}} \\
= & \qquad \langle \ \varphi \in D_J \ \rangle \\
& \{(i,A) \mid i \in J \ \wedge \ A = \varphi(i)\}^{\downarrow_{D_K}} \\
= & \qquad \langle \text{ Definition of } \downarrow_{D_K} \rangle \\
& \mathcal{I}_K;\{(i,A) \mid i \in J \ \wedge \ A = \varphi(i)\} \\
= & \qquad \langle \text{ Proposition 1(1) } \rangle \\
& \{(i,A) \mid i \in J \cap K \ \wedge \ A = \varphi(i)\} \\
= & \qquad \langle \text{ Set theory } \rangle \\
& \{(i,A) \mid i \in J \cap (J \cap K) \ \wedge \ A = \varphi(i)\} \\
= & \qquad \langle \text{ Definition of } \downarrow_{D_K} \rangle \\
& \varphi^{\downarrow_{D_{J \cap K}}}
\end{array}
$$

4.

$$
\begin{array}{ll}
& (\varphi^{\downarrow_{D_K}})^{\downarrow_{D_L}} \\
= & \qquad \langle \text{ Definition of } \downarrow_{D_L} \rangle \\
& \mathcal{I}_L;(\varphi^{\downarrow_{D_K}}) \\
= & \qquad \langle \text{ Definition of } \downarrow_{D_K} \rangle \\
& \mathcal{I}_K;\mathcal{I}_L;\varphi \\
= & \qquad \langle \text{ Proposition 1(2) } \rangle \\
& \mathcal{I}_{K \cap L};\varphi \\
= & \qquad \langle \text{ Definition of } \downarrow_{D_{K \cap L}} \rangle \\
& \varphi^{\downarrow_{D_{K \cap L}}}
\end{array}
$$

□

A.4 Proposition 15

1. $D_J \preccurlyeq d(\varphi) \ \vee \ \varphi = fs(\varphi, D_J, D_K)$
2. $D_K \preccurlyeq d(\varphi)) \ \vee \ \varphi = fs(fs(\varphi, D_J, D_K), D_K, D_J)$

Proof. 1.

$$
\begin{array}{ll}
& D_J \preccurlyeq d(\varphi) \ \vee \ \varphi = fs(\varphi, D_J, D_K) \\
\leftrightarrow & \qquad \langle \ p \vee q \leftrightarrow \neg p \rightarrow q \ \rangle \\
& \neg(D_J \preccurlyeq d(\varphi)) \rightarrow \varphi = fs(\varphi, D_J, D_K)
\end{array}
$$

Our proof strategy for $p \to q$ is to assume p and then prove q. We assume

$$
\begin{array}{ll}
 & \neg(D_J \preccurlyeq d(\varphi)) \\
\leftrightarrow & \quad \langle\ d(\varphi) = D_L\ \rangle \\
 & \neg(D_J \preccurlyeq D_L) \\
\to & \quad \langle\ \text{Proposition 7(3)}\ \rangle \\
 & \neg(J \subseteq L) \\
\to & \quad \langle\ \text{Proposition 14}\ \rangle \\
 & J \cap L = \emptyset
\end{array}
$$

Then we prove $fs(\varphi, D_J, D_K) = \varphi$

$$
\begin{array}{ll}
 & fs(\varphi, D_J, D_K) \\
= & \quad \langle\ \text{Definition of } fs(\varphi, D_J, D_K)\ \rangle \\
 & \varphi^{\downarrow_{D_{(L-J)}}} \cdot (\varphi^{\downarrow_{D_J}}[D_K/D_J]) \\
= & \quad \langle\ (J \cap L = \emptyset) \to L - J = L\ \rangle \\
 & \varphi^{\downarrow_{D_L}} \cdot (\varphi^{\downarrow_{D_J}}[D_K/D_J]) \\
= & \quad \langle\ \text{Proposition 10(1)}\ \rangle \\
 & \varphi \cdot (\varphi^{\downarrow_{D_J}}[D_K/D_J]) \\
= & \quad \langle\ \varphi \in D_L\ \rangle \\
 & \varphi \cdot (\{(i,A)\ \mid\ i \in L\ \wedge\ A = \varphi(i)\}^{\downarrow_{D_J}}[D_K/D_J]) \\
= & \quad \langle\ \text{Definition of } \downarrow_{D_J}\ \rangle \\
 & \varphi \cdot (\mathcal{I}_J;\{(i,A)\ \mid\ i \in L\ \wedge\ A = \varphi(i)\}[D_K/D_J]) \\
= & \quad \langle\ \text{Proposition 1(1)}\ \rangle \\
 & \varphi \cdot (\{(i,A)\ \mid\ i \in L \cap J\ \wedge\ A = \varphi(i)\}[D_K/D_J]) \\
= & \quad \langle\ J \cap L = \emptyset\ \rangle \\
 & \varphi \cdot (\{(i,A)\ \mid\ i \in \emptyset\ \wedge\ A = \varphi(i)\}[D_K/D_J]) \\
= & \quad \langle\ i \in \emptyset \leftrightarrow \mathsf{false}\ \rangle \\
 & \varphi \cdot (\{(i,A)\ \mid\ \mathsf{false}\ \wedge\ A = \varphi(i)\}[D_K/D_J]) \\
= & \quad \langle\ p\ \wedge\ \mathsf{false} \leftrightarrow \mathsf{false}\ \rangle \\
 & \varphi \cdot (\{(i,A)\ \mid\ \mathsf{false}\}[D_K/D_J]) \\
= & \quad \langle\ \text{Empty range}\ \rangle \\
 & \varphi \cdot (\emptyset[D_K/D_J]) \\
= & \quad \langle\ \text{Frame substitution of an empty set}\ \rangle \\
 & \varphi \cdot \emptyset \\
= & \quad \langle\ \text{Proposition 11(4)}\ \rangle \\
 & \varphi \cdot e_\emptyset \\
= & \quad \langle\ \text{Proposition 11(3)}\ \rangle \\
 & \varphi
\end{array}
$$

2.
$$
\begin{array}{ll}
 & D_K \preccurlyeq d(\varphi)\ \ \vee\ \ \varphi = fs(fs(\varphi, D_J, D_K), D_K, D_J) \\
\leftrightarrow & \quad \langle\ p \vee q \leftrightarrow \neg p \to q\ \rangle \\
 & \neg(D_K \preccurlyeq d(\varphi)) \to \varphi = fs(fs(\varphi, D_J, D_K), D_K, D_J)
\end{array}
$$

Our proof strategy for $p \to q$ is to assume p and then prove q. We assume

$$\neg(D_K \preccurlyeq d(\varphi))$$

$$
\begin{array}{ll}
\leftrightarrow & \langle\ d(\varphi) = D_L\ \rangle \\
\quad \neg(D_K \preccurlyeq D_L) & \\
\rightarrow & \langle\ \text{Proposition 7(3)}\ \rangle \\
\quad \neg(K \subseteq L) & \\
\rightarrow & \langle\ \text{Proposition 14}\ \rangle \\
\quad K \cap L = \emptyset &
\end{array}
$$

To prove $\varphi = fs(fs(\varphi, D_J, D_K), D_K, D_J)$, we have two cases:

(a) $\neg(D_J \preccurlyeq d(\varphi))$

(b) $D_J \preccurlyeq d(\varphi)$

(a) Assume $\neg(D_J \preccurlyeq d(\varphi))$

$$
\begin{array}{ll}
 & fs(fs(\varphi, D_J, D_K), D_K, D_J) \\
= & \langle\ \text{Proposition 15(1) \& } \neg D_J \preccurlyeq d(\varphi)\ \rangle \\
 & fs(\varphi, D_K, D_J) \\
= & \langle\ \text{Proposition 15(1) \& } \neg D_K \preccurlyeq d(\varphi)\ \rangle \\
 & \varphi
\end{array}
$$

(b) Assume $D_J \preccurlyeq d(\varphi) \rightarrow J \subseteq L \rightarrow J \cap L = J$

$$
\begin{array}{ll}
 & fs(fs(\varphi, D_J, D_K), D_K, D_J) \\
= & \langle\ \text{Definition of } fs\ \rangle \\
 & fs(\varphi^{\downarrow D_{(L-J)}} \cdot (\varphi^{\downarrow D_J}[D_K/D_J]), D_K, D_J) \\
= & \langle\ \text{Definition of } fs\ \rangle \\
 & (\varphi^{\downarrow D_{(L-J)}} \cdot (\varphi^{\downarrow D_J}[D_K/D_J]))^{\downarrow D_{(L-K)}} \cdot ((\varphi^{\downarrow D_{(L-J)}} \cdot (\varphi^{\downarrow D_J}[D_K/D_J]))^{\downarrow D_K}[D_J/D_K]) \\
= & \langle\ (K \cap L = \emptyset) \rightarrow L - K = L\ \rangle \\
 & (\varphi^{\downarrow D_{(L-J)}} \cdot (\varphi^{\downarrow D_J}[D_K/D_J]))^{\downarrow D_L} \cdot ((\varphi^{\downarrow D_{(L-J)}} \cdot (\varphi^{\downarrow D_J}[D_K/D_J]))^{\downarrow D_K}[D_J/D_K]) \\
= & \langle\ \text{Proposition 1(2, 9) \& } d(\varphi^{\downarrow D_J}[D_K/D_J]) = D_K\ \rangle \\
 & (\varphi^{\downarrow D_{(L-J)}} \cdot (\varphi^{\downarrow D_J}[D_K/D_J]))^{\downarrow D_L} \cdot ((\varphi^{\downarrow D_{(L-J)\cap K}} \cdot (\varphi^{\downarrow D_J}[D_K/D_J]))[D_J/D_K]) \\
= & \langle\ K \cap L = \emptyset\ \rangle \\
 & (\varphi^{\downarrow D_{(L-J)}} \cdot (\varphi^{\downarrow D_J}[D_K/D_J]))^{\downarrow D_L} \cdot ((\varphi^{\downarrow D_\emptyset} \cdot (\varphi^{\downarrow D_J}[D_K/D_J]))[D_J/D_K]) \\
= & \langle\ \text{Proposition 11(1)}\ \rangle \\
 & (\varphi^{\downarrow D_{(L-J)}} \cdot (\varphi^{\downarrow D_J}[D_K/D_J]))^{\downarrow D_L} \cdot ((e_\emptyset \cdot (\varphi^{\downarrow D_J}[D_K/D_J]))[D_J/D_K]) \\
= & \langle\ \text{Proposition 11(3)}\ \rangle \\
 & (\varphi^{\downarrow D_{(L-J)}} \cdot (\varphi^{\downarrow D_J}[D_K/D_J]))^{\downarrow D_L} \cdot ((\varphi^{\downarrow D_J}[D_K/D_J])[D_J/D_K]) \\
= & \langle\ \text{Replace } J \text{ by } K \text{ and then } K \text{ by } J \text{ is equivalent to replace } J \text{ by } J\ \rangle \\
 & (\varphi^{\downarrow D_{(L-J)}} \cdot (\varphi^{\downarrow D_J}[D_K/D_J]))^{\downarrow D_L} \cdot (\varphi^{\downarrow D_J}[D_J/D_J]) \\
= & \langle\ \text{Replacing } J \text{ by } J \text{ does not affect the information}\ \rangle \\
 & (\varphi^{\downarrow D_{(L-J)}} \cdot (\varphi^{\downarrow D_J}[D_K/D_J]))^{\downarrow D_L} \cdot \varphi^{\downarrow D_J} \\
= & \langle\ \text{Proposition 12(2)}\ \rangle \\
 & (\varphi^{\downarrow D_{(L-J)}})^{\downarrow D_L} \cdot (\varphi^{\downarrow D_J}[D_K/D_J])^{\downarrow D_L} \cdot \varphi^{\downarrow D_J} \\
= & \langle\ \text{Proposition 12(4)}\ \rangle \\
 & (\varphi^{\downarrow D_{(L-J)\cap L}}) \cdot (\varphi^{\downarrow D_J}[D_K/D_J])^{\downarrow D_L} \cdot \varphi^{\downarrow D_J}
\end{array}
$$

$$
\begin{array}{ll}
= & \langle\ (L-J)\cap L = L\cap\overline{J}\cap L = L\cap\overline{J} = L-J\ \rangle \\
& (\varphi^{\downarrow D_{(L-J)}})\cdot(\varphi^{\downarrow D_J}[D_K/D_J])^{\downarrow D_L}\cdot\varphi^{\downarrow D_J} \\
= & \langle\ \text{Proposition 12(3) \& } d(\varphi^{\downarrow D_J}[D_K/D_J]) = D_K\ \rangle \\
& (\varphi^{\downarrow D_{(L-J)}})\cdot(\varphi^{\downarrow D_J}[D_K/D_J])^{\downarrow D_{L\cap K}}\cdot\varphi^{\downarrow D_J} \\
= & \langle\ K\cap L = \emptyset\ \rangle \\
& (\varphi^{\downarrow D_{(L-J)}})\cdot(\varphi^{\downarrow D_J}[D_K/D_J])^{\downarrow D_\emptyset}\cdot\varphi^{\downarrow D_J} \\
= & \langle\ \text{Proposition 11(1)}\ \rangle \\
& (\varphi^{\downarrow D_{(L-J)}})\cdot e_\emptyset\cdot\varphi^{\downarrow D_J} \\
= & \langle\ \text{Proposition 11(3)}\ \rangle \\
& (\varphi^{\downarrow D_{(L-J)}})\cdot\varphi^{\downarrow D_J} \\
= & \langle\ \text{Proposition 12(1)}\ \rangle \\
& \varphi^{\downarrow D_{(L-J)\cup J}} \\
= & \langle\ D_J \preccurlyeq d(\varphi) \rightarrow J\subseteq L\ \&\ (L-J)\cup J = L\ \rangle \\
& \varphi^{\downarrow D_L} \\
= & \langle\ \text{Proposition 12(1)}\ \rangle \\
& \varphi
\end{array}
$$

□

A.5 Proposition 16

1. $\varphi\leq\psi\ \wedge\ \varphi\leq\chi \rightarrow update(update(\mathcal{N},\varphi,\psi),\varphi,\chi) = update(\mathcal{N},\varphi,\psi\cdot\chi)$
2. $isInKnowledge(\mathcal{N}, d(\varphi),\varphi)\ \vee\ update(\mathcal{N},\varphi,\psi) = \mathcal{N}$

Proof. 1.

$$
\begin{array}{ll}
& \Psi \\
& \qquad \langle\ \text{Definition of the function update}\ \rangle \\
= & \{\tau\mid\exists(\chi_1\mid\chi_1\in\Omega:\varphi\leq\chi_1\ \wedge\ \tau=(\chi_1-\varphi)\cdot\chi)\} \\
& \cup\ \{\tau\mid\exists(\chi_1\mid\chi_1\in\Omega:\neg(\varphi\leq\chi_1)\ \wedge\ \tau=\chi_1)\} \\
& \qquad \langle\ \text{Definition of the function update}\ \rangle \\
= & \{\tau\mid\exists(\chi_1\mid\chi_1\in \\
& \qquad \{\tau_1\mid\exists(\chi_2\mid\chi_2\in\Phi:\varphi\leq\chi_2\ \wedge\ \tau_1=(\chi_2-\varphi)\cdot\psi)\} \\
& \qquad \cup\ \{\tau_1\mid\exists(\chi_2\mid\chi_2\in\Phi:\neg(\varphi\leq\chi_2)\ \wedge\ \tau_1=\chi_2)\}: \\
& \qquad \varphi\leq\chi_1\ \wedge\ \tau=(\chi_1-\varphi)\cdot\chi)\} \\
& \cup\ \{\tau\mid\exists(\chi_1\mid\chi_1\in \\
& \qquad \{\tau_1\mid\exists(\chi_2\mid\chi_2\in\Phi:\varphi\leq\chi_2\ \wedge\ \tau_1=(\chi_2-\varphi)\cdot\psi)\} \\
& \qquad \cup\ \{\tau_1\mid\exists(\chi_2\mid\chi_2\in\Phi:\neg(\varphi\leq\chi_2)\ \wedge\ \tau_1=\chi_2)\}: \\
& \qquad \neg(\varphi\leq\chi_1)\ \wedge\ \tau=\chi_1)\} \\
= & \qquad \langle\ \text{Set union axiom (i.e., } i\in A\vee i\in B\leftrightarrow i\in A\cup B)\ \rangle \\
& \{\tau\mid\exists(\chi_1\mid \\
& \qquad \chi_1\in\{\tau_1\mid\exists(\chi_2\mid\chi_2\in\Phi:\varphi\leq\chi_2\ \wedge\ \tau_1=(\chi_2-\varphi)\cdot\psi)\} \\
& \qquad \vee\ \chi_1\in\{\tau_1\mid\exists(\chi_2\mid\chi_2\in\Phi:\neg(\varphi\leq\chi_2)\ \wedge\ \tau_1=\chi_2)\}: \\
& \qquad \varphi\leq\chi_1\ \wedge\ \tau=(\chi_1-\varphi)\cdot\chi)\} \\
& \cup\ \{\tau\mid\exists(\chi_1\mid \\
& \qquad \chi_1\in\{\tau_1\mid\exists(\chi_2\mid\chi_2\in\Phi:\varphi\leq\chi_2\ \wedge\ \tau_1=(\chi_2-\varphi)\cdot\psi)\} \\
& \qquad \vee\ \chi_1\in\{\tau_1\mid\exists(\chi_2\mid\chi_2\in\Phi:\neg(\varphi\leq\chi_2)\ \wedge\ \tau_1=\chi_2)\}:
\end{array}
$$

$$
\begin{array}{ll}
 & \neg(\varphi \le \chi_1) \ \wedge \ \tau = \chi_1)\} \\
= & \qquad \langle\ y \in \{x \mid r\} \leftrightarrow r[x := y]\ \rangle \\
 & \{\tau \mid \exists(\chi_1 \mid \\
 & \quad \exists(\chi_2 \mid \chi_2 \in \Phi : \varphi \le \chi_2 \ \wedge \ \chi_1 = (\chi_2 - \varphi)\cdot\psi) \\
 & \quad \vee \ \exists(\chi_2 \mid \chi_2 \in \Phi : \neg(\varphi \le \chi_2) \ \wedge \ \chi_1 = \chi_2) : \\
 & \quad \varphi \le \chi_1 \ \wedge \ \tau = (\chi_1 - \varphi)\cdot\chi)\} \\
 & \cup \{\tau \mid \exists(\chi_1 \mid \\
 & \quad \exists(\chi_2 \mid \chi_2 \in \Phi : \varphi \le \chi_2 \ \wedge \ \chi_1 = (\chi_2 - \varphi)\cdot\psi) \\
 & \quad \vee \ \exists(\chi_2 \mid \chi_2 \in \Phi : \neg(\varphi \le \chi_2) \ \wedge \ \chi_1 = \chi_2) : \\
 & \quad \neg(\varphi \le \chi_1) \ \wedge \ \tau = \chi_1)\} \\
= & \qquad \langle\ \text{Distributivity axiom}\ \rangle \\
 & \{\tau \mid \exists(\chi_1 \mid \ \exists(\chi_2 \mid \chi_2 \in \Phi : \\
 & \quad (\varphi \le \chi_2 \ \wedge \ \chi_1 = (\chi_2 - \varphi)\cdot\psi) \ \vee \ (\neg(\varphi \le \chi_2) \ \wedge \ \chi_1 = \chi_2)) : \\
 & \quad \varphi \le \chi_1 \ \wedge \ \tau = (\chi_1 - \varphi)\cdot\chi)\} \\
 & \cup \{\tau \mid \exists(\chi_1 \mid \ \exists(\chi_2 \mid \chi_2 \in \Phi : \\
 & \quad (\varphi \le \chi_2 \ \wedge \ \chi_1 = (\chi_2 - \varphi)\cdot\psi) \ \vee \ (\neg(\varphi \le \chi_2) \ \wedge \ \chi_1 = \chi_2)) : \\
 & \quad \neg(\varphi \le \chi_1) \ \wedge \ \tau = \chi_1)\} \\
= & \qquad \langle\ \text{Trading rule for}\ \exists\ \rangle \\
 & \{\tau \mid \exists(\chi_1 \mid : \ \exists(\chi_2 \mid \chi_2 \in \Phi : \\
 & \quad (\varphi \le \chi_2 \ \wedge \ \chi_1 = (\chi_2 - \varphi)\cdot\psi) \ \vee \ (\neg(\varphi \le \chi_2) \ \wedge \ \chi_1 = \chi_2)) \\
 & \quad \wedge \ \varphi \le \chi_1 \ \wedge \ \tau = (\chi_1 - \varphi)\cdot\chi)\} \\
 & \cup \{\tau \mid \exists(\chi_1 \mid : \ \exists(\chi_2 \mid \chi_2 \in \Phi : \\
 & \quad (\varphi \le \chi_2 \ \wedge \ \chi_1 = (\chi_2 - \varphi)\cdot\psi) \ \vee \ (\neg(\varphi \le \chi_2) \ \wedge \ \chi_1 = \chi_2)) \\
 & \quad \wedge \ \neg(\varphi \le \chi_1) \ \wedge \ \tau = \chi_1)\} \\
= & \qquad \langle\ \text{Nesting axiom}\ \rangle \\
 & \{\tau \mid \exists(\chi_1,\chi_2 \mid \chi_2 \in \Phi : \\
 & \quad \Big((\varphi \le \chi_2 \ \wedge \ \chi_1 = (\chi_2 - \varphi)\cdot\psi) \ \vee \ (\neg(\varphi \le \chi_2) \ \wedge \ \chi_1 = \chi_2)\Big) \\
 & \quad \wedge \ (\varphi \le \chi_1 \ \wedge \ \tau = (\chi_1 - \varphi)\cdot\chi))\} \\
 & \cup \{\tau \mid \exists(\chi_1,\chi_2 \mid \chi_2 \in \Phi : \\
 & \quad \Big((\varphi \le \chi_2 \ \wedge \ \chi_1 = (\chi_2 - \varphi)\cdot\psi) \ \vee \ (\neg(\varphi \le \chi_2) \ \wedge \ \chi_1 = \chi_2)\Big) \\
 & \quad \wedge \ (\neg(\varphi \le \chi_1) \ \wedge \ \tau = \chi_1))\} \\
= & \qquad \langle\ \text{Distributivity of}\ \wedge\ \text{over}\ \vee\ \rangle \\
 & \{\tau \mid \exists(\chi_1,\chi_2 \mid \chi_2 \in \Phi : \\
 & \quad \Big(\varphi \le \chi_2 \ \wedge \ \chi_1 = (\chi_2 - \varphi)\cdot\psi \ \wedge \ \varphi \le \chi_1 \ \wedge \ \tau = (\chi_1 - \varphi)\cdot\chi\Big) \\
 & \quad \vee \ \Big(\neg(\varphi \le \chi_2) \ \wedge \ \chi_1 = \chi_2 \ \wedge \ \varphi \le \chi_1 \ \wedge \ \tau = (\chi_1 - \varphi)\cdot\chi\Big))\} \\
 & \cup \{\tau \mid \exists(\chi_1,\chi_2 \mid \chi_2 \in \Phi : \\
 & \quad \Big(\varphi \le \chi_2 \ \wedge \ \chi_1 = (\chi_2 - \varphi)\cdot\psi) \ \wedge \ \neg(\varphi \le \chi_1) \ \wedge \ \tau = \chi_1\Big) \\
 & \quad \vee \ \Big(\neg(\varphi \le \chi_2) \ \wedge \ \chi_1 = \chi_2 \ \wedge \ \neg(\varphi \le \chi_1) \ \wedge \ \tau = \chi_1\Big))\} \\
= & \qquad \langle\ \text{Proposition 13(7)}\ \rangle \\
 & \{\tau \mid \exists(\chi_1,\chi_2 \mid \chi_2 \in \Phi : \\
 & \quad \Big(\varphi \le \chi_2 \ \wedge \ \chi_1 = \chi_2\cdot\psi \ \wedge \ \varphi \le \chi_1 \ \wedge \ \tau = \chi_1\cdot\chi\Big) \\
 & \quad \vee \ \Big(\neg(\varphi \le \chi_2) \ \wedge \ \chi_1 = \chi_2 \ \wedge \ \varphi \le \chi_1 \ \wedge \ \tau = \chi_1\cdot\chi\Big))\} \\
 & \cup \{\tau \mid \exists(\chi_1,\chi_2 \mid \chi_2 \in \Phi :
\end{array}
$$

$$
\begin{aligned}
&\Big(\varphi \leq \chi_2 \ \wedge\ \chi_1 = \chi_2 \cdot \psi \ \wedge\ \neg(\varphi \leq \chi_1) \ \wedge\ \tau = \chi_1\Big) \\
&\quad \vee \ \Big(\neg(\varphi \leq \chi_2) \ \wedge\ \chi_1 = \chi_2 \ \wedge\ \neg(\varphi \leq \chi_1) \ \wedge\ \tau = \chi_1\Big))\} \\
= &\quad \langle \text{ Substitution axiom } \rangle \\
&\{\tau \mid \exists(\chi_1, \chi_2 \mid \chi_2 \in \Phi : \\
&\quad \Big(\varphi \leq \chi_2 \ \wedge\ \chi_1 = \chi_2 \cdot \psi \ \wedge\ \varphi \leq \chi_2 \cdot \psi \ \wedge\ \tau = \chi_2 \cdot \psi \cdot \chi\Big) \\
&\quad \vee \ \Big(\neg(\varphi \leq \chi_2) \ \wedge\ \chi_1 = \chi_2 \ \wedge\ \varphi \leq \chi_2 \ \wedge\ \tau = \chi_2 \cdot \chi\Big))\} \\
&\cup \{\tau \mid \exists(\chi_1, \chi_2 \mid \chi_2 \in \Phi : \\
&\quad \Big(\varphi \leq \chi_2 \ \wedge\ \chi_1 = \chi_2 \cdot \psi \ \wedge\ \neg(\varphi \leq \chi_2 \cdot \psi) \ \wedge\ \tau = \chi_2 \cdot \psi\Big) \\
&\quad \vee \ \Big(\neg(\varphi \leq \chi_2) \ \wedge\ \chi_1 = \chi_2 \ \wedge\ \neg(\varphi \leq \chi_2) \ \wedge\ \tau = \chi_2\Big))\} \\
= &\quad \langle \text{ Contradiction } \rangle \\
&\{\tau \mid \exists(\chi_1, \chi_2 \mid \chi_2 \in \Phi : \\
&\quad \Big(\varphi \leq \chi_2 \ \wedge\ \chi_1 = \chi_2 \cdot \psi \ \wedge\ \varphi \leq \chi_2 \cdot \psi \ \wedge\ \tau = \chi_2 \cdot \psi \cdot \chi\Big) \\
&\quad \vee \ \mathsf{false})\} \\
&\cup \{\tau \mid \exists(\chi_1, \chi_2 \mid \chi_2 \in \Phi : \\
&\quad \mathsf{false} \\
&\quad \vee \ \Big(\neg(\varphi \leq \chi_2) \ \wedge\ \chi_1 = \chi_2 \ \wedge\ \neg(\varphi \leq \chi_2) \ \wedge\ \tau = \chi_2\Big))\} \\
= &\quad \langle \text{ Zero for } \vee \rangle \\
&\{\tau \mid \exists(\chi_1, \chi_2 \mid \chi_2 \in \Phi : \\
&\quad (\varphi \leq \chi_2 \ \wedge\ \chi_1 = \chi_2 \cdot \psi \ \wedge\ \varphi \leq \chi_2 \cdot \psi \ \wedge\ \tau = \chi_2 \cdot \psi \cdot \chi))\} \\
&\cup \{\tau \mid \exists(\chi_1, \chi_2 \mid \chi_2 \in \Phi : \\
&\quad (\neg(\varphi \leq \chi_2) \ \wedge\ \chi_1 = \chi_2 \ \wedge\ \neg(\varphi \leq \chi_2) \ \wedge\ \tau = \chi_2))\} \\
= &\quad \langle\ \varphi \leq \chi_2 \rightarrow \varphi \leq \chi_2 \cdot \psi\ \rangle \\
&\{\tau \mid \exists(\chi_1, \chi_2 \mid \chi_2 \in \Phi : \\
&\quad (\varphi \leq \chi_2 \ \wedge\ \chi_1 = \chi_2 \cdot \psi \ \wedge\ \tau = \chi_2 \cdot \psi \cdot \chi))\} \\
&\cup \{\tau \mid \exists(\chi_1, \chi_2 \mid \chi_2 \in \Phi : \\
&\quad (\neg(\varphi \leq \chi_2) \ \wedge\ \chi_1 = \chi_2 \ \wedge\ \neg(\varphi \leq \chi_2) \ \wedge\ \tau = \chi_2))\} \\
= &\quad \langle \text{ Idempotency of } \wedge\ \rangle \\
&\{\tau \mid \exists(\chi_1, \chi_2 \mid \chi_2 \in \Phi : \\
&\quad (\varphi \leq \chi_2 \ \wedge\ \chi_1 = \chi_2 \cdot \psi \ \wedge\ \tau = \chi_2 \cdot \psi \cdot \chi))\} \\
&\cup \{\tau \mid \exists(\chi_1, \chi_2 \mid \chi_2 \in \Phi : \\
&\quad (\neg(\varphi \leq \chi_2) \ \wedge\ \chi_1 = \chi_2 \ \wedge\ \tau = \chi_2))\},\ D) \\
= &\quad \langle \text{ Trading rule for } \exists \text{ \& Symmetry of } \wedge\ \rangle \\
&(\{\tau \mid \exists(\chi_1, \chi_2 \mid \chi_2 \in \Phi \ \wedge\ \varphi \leq \chi_2 \ \wedge\ \tau = \chi_2 \cdot \psi \cdot \chi) : \\
&\quad (\chi_1 = \chi_2 \cdot \psi)\} \\
&\cup \{\tau \mid \exists(\chi_1, \chi_2 \mid \chi_2 \in \Phi \ \wedge\ \neg(\varphi \leq \chi_2) \ \wedge\ \tau = \chi_2) : \\
&\quad (\chi_1 = \chi_2)\} \\
= &\quad \langle \text{ Nesting axiom } \rangle \\
&\{\tau \mid \exists(\chi_2 \mid \chi_2 \in \Phi \ \wedge\ \varphi \leq \chi_2 \ \wedge\ \tau = \chi_2 \cdot \psi \cdot \chi) : \\
&\quad \exists(\chi_1 \mid\!: (\chi_1 = \chi_2 \cdot \psi))\} \\
&\cup \{\tau \mid \exists(\chi_2 \mid \chi_2 \in \Phi \ \wedge\ \neg(\varphi \leq \chi_2) \ \wedge\ \tau = \chi_2) : \\
&\quad \exists(\chi_1 \mid\!: (\chi_1 = \chi_2))\} \\
= &\quad \langle \text{ The information } \chi_1 \text{ and combining information is always defined } \rangle \\
&\{\tau \mid \exists(\chi_2 \mid \chi_2 \in \Phi \ \wedge\ \varphi \leq \chi_2 \ \wedge\ \tau = \chi_2 \cdot \psi \cdot \chi) :
\end{aligned}
$$

$$
\begin{array}{ll}
 & \quad \mathsf{true}\,)\} \\
 & \cup\ \{\tau \mid\ \exists(\chi_2 \mid \chi_2 \in \Phi \ \wedge\ \neg(\varphi \leq \chi_2)\ \wedge\ \tau = \chi_2)\ : \\
 & \quad \mathsf{true}\,)\} \\
= & \qquad \langle\ \text{Trading rule for } \exists\ \rangle \\
 & \quad \{\tau \mid\ \exists(\chi_2 \mid \chi_2 \in \Phi\ :\ \varphi \leq \chi_2\ \wedge\ \tau = \chi_2 \cdot \psi \cdot \chi)\} \\
 & \cup\ \{\tau \mid\ \exists(\chi_2 \mid \chi_2 \in \Phi\ :\ \neg(\varphi \leq \chi_2)\ \wedge\ \tau = \chi_2))\} \\
= & \qquad \langle\ \text{Proposition 13(7)}\ \rangle \\
 & \quad \{\tau \mid\ \exists(\chi_2 \mid \chi_2 \in \Phi\ :\ \varphi \leq \chi_2\ \wedge\ \tau = (\chi_2 - \varphi) \cdot \psi \cdot \chi)\} \\
 & \cup\ \{\tau \mid\ \exists(\chi_2 \mid \chi_2 \in \Phi\ :\ \neg(\varphi \leq \chi_2)\ \wedge\ \tau = \chi_2))\} \\
= & \qquad \langle\ \text{The definition of } \Omega\ \rangle \\
 & \Omega[\psi \cdot \chi / \psi]
\end{array}
$$

2.

$$
\begin{array}{ll}
 & \mathit{isInKnowledge}(\mathcal{N},\ d(\varphi),\ \varphi)\ \ \vee\ \ \mathit{update}(\mathcal{N},\ \varphi,\ \psi) = \mathcal{N} \\
\leftrightarrow & \qquad \langle\ p \rightarrow q \leftrightarrow \neg p \vee q\ \rangle \\
 & \neg \mathit{isInKnowledge}(\mathcal{N},\ d(\varphi),\ \varphi) \rightarrow \mathit{update}(\mathcal{N},\ \varphi,\ \psi) = \mathcal{N}
\end{array}
$$

First, we assume that

$$
\begin{array}{ll}
 & \neg \mathit{isInKnowledge}(\mathcal{N},\ d(\varphi),\ \varphi) \\
\leftrightarrow & \qquad \langle\ \text{Definition of } \mathit{isInKnowledge}\ \rangle \\
 & \neg\ \exists(\chi \mid\ \chi \in \Phi\ :\ d(\varphi) \in D\ \ \wedge\ \ \varphi \leq \chi\ \ \wedge\ \ d(\varphi) \preccurlyeq d(\chi))\ ) \\
\leftrightarrow & \qquad \langle\ \text{De Morgan}\ \rangle \\
 & \forall(\chi \mid\ \chi \in \Phi\ :\ \neg(d(\varphi) \in D\ \ \wedge\ \ \varphi \leq \psi\ \ \wedge\ \ d(\varphi) \preccurlyeq d(\chi)))\ ) \\
\leftrightarrow & \qquad \langle\ \text{De Morgan}\ \rangle \\
 & \forall(\chi \mid\ \chi \in \Phi\ :\ \neg(d(\varphi) \in D)\ \ \vee\ \ \neg(\varphi \leq \chi)\ \ \vee\ \ (\neg d(\varphi) \preccurlyeq d(\chi)))\ )
\end{array}
$$

Based on the assumption, we prove that for $d(\varphi) = D_J$ and $d(\psi) = D_K$, we have $\exists(\chi \mid \chi \in \Phi\ :\ \varphi \leq \chi) \rightarrow \mathsf{false}$

$$
\begin{array}{ll}
 & \exists(\chi \mid\ \chi \in \Phi\ :\ \varphi \leq \chi) \\
\rightarrow & \qquad \langle\ \varphi \leq \chi \rightarrow J \subseteq K \text{ from the definition of } \leq\ \rangle \\
 & \exists(\chi \mid\ \chi \in \Phi\ :\ \varphi \leq \chi\ \wedge\ J \subseteq K) \\
\rightarrow & \qquad \langle\ \text{Proposition 7(1)}\ \rangle \\
 & \exists(\chi \mid\ \chi \in \Phi\ :\ \varphi \leq \chi\ \wedge\ D_J \preccurlyeq D_K) \\
\rightarrow & \qquad \langle\ d(\varphi) = D_J \text{ and } d(\psi) = D_K\ \rangle \\
 & \exists(\chi \mid\ \chi \in \Phi\ :\ \varphi \leq \chi\ \wedge\ d(\varphi) \preccurlyeq d(\chi)) \\
\rightarrow & \qquad \langle\ d(\chi) \in D\ \wedge\ J \subseteq K \rightarrow d(\varphi) \in D\ \rangle \\
 & \exists(\chi \mid\ \chi \in \Phi\ :\ \varphi \leq \chi\ \wedge\ d(\varphi) \preccurlyeq d(\chi)\ \wedge\ d(\varphi) \in D) \\
\rightarrow & \qquad \langle\ \text{The assumption}\ \rangle \\
 & \mathsf{false}
\end{array}
$$

Therefore, we have $\neg\ \exists(\chi \mid\ \chi \in \Phi\ :\ \varphi \leq \chi) \leftrightarrow\ \forall(\chi \mid\ \chi \in \Phi\ :\ \neg(\varphi \leq \chi)) \leftrightarrow \mathsf{true}$. Then we prove that $\mathit{update}(\mathcal{N},\ \varphi,\ \psi) = \mathcal{N}$

$$
\begin{array}{ll}
 & \mathit{update}(\mathcal{N},\ \varphi,\ \psi) \\
= & \qquad \langle\ \text{Definition of } \mathit{update}\ \rangle \\
 & (\{(\chi - \varphi) \cdot \psi \mid\ \chi \in \Phi\ \wedge\ \varphi \leq \chi\}\ \cup\ \{\chi \mid\ \chi \in \Phi\ \wedge\ \neg(\varphi \leq \chi)\},\ D)
\end{array}
$$

$$
\begin{array}{ll}
= & \langle \text{ From (1) } \rangle \\
& (\{(\chi - \varphi) \cdot \psi \mid \chi \in \Phi \wedge \mathsf{false}\} \cup \{\chi \mid \chi \in \Phi \wedge \mathsf{true}\}, D) \\
= & \langle \text{ Zero of } \wedge \text{ and } \vee \rangle \\
& (\{(\chi - \varphi) \cdot \psi \mid \mathsf{false}\} \cup \{\chi \mid \chi \in \Phi\}, D) \\
= & \langle \text{ Empty range } \rangle \\
& (\emptyset \cup \{\chi \mid \chi \in \Phi\}, D) \\
= & \langle \text{ Identity of } \cup \rangle \\
& (\{\chi \mid \chi \in \Phi\}, D) \\
= & \langle \text{ Definition of } \mathcal{N} \rangle \\
& \mathcal{N}
\end{array}
$$

□

B. References

Alghathbar, K., Farkas, C., and Wijesekera, D. (2006). Securing UML information flow using FlowUML. *Journal of Research and Practice in Information Technology*, 38(1):111–120.

Bell, D. and LaPadula, L. (1976). Secure computer system: Unified exposition and multics interpretation. Technical Report ESD-TR-75-306, The MITRE Corporation.

Brewer, D. F. and Nash, M. J. (1989). The chinese wall security policy. In *IEEE Symposium on Security and Privacy*, pages 206–214.

Cervesato, I. (2000). Typed multiset rewriting specifications of security protocols. In Seda, A., editor, *First Irish Conference on the Mathematical Foundations of Computer Science and Information Technology — MFCSIT'00*, pages 1–43, Cork, Ireland. Elsevier ENTCS 40.

Clarke, E. M., Jha, S., and Marrero, W. (2000). Verifying Security Protocols with Brutus. *ACM Transactions on Software Engineering and Methodology*, 9(4):443–487.

Davey, B. and Priestley, H. (2002). *Introduction to Lattices and Order*. Cambridge university press, second edition.

Fábrega, F. J. T., Herzog, J. C., and Guttman, J. D. (1999). Strand Spaces: Proving Security Protocols Correct. *Journal of Computer Security*, 7(2–3):191–230.

Focardi, R. and Gorrieri, R. (1997). The Compositional Security Checker: A Tool for the Verification of Information Flow Security Properties. *IEEE Transactions on Software Engineering*, 23(9):550–571.

Hristova, K., Rothamel, T., Liu, Y. A., and Stoller, S. D. (2006). Efficient type inference for secure information flow. In *PLAS '06: Proceedings of the 2006 workshop on Programming languages and analysis for security*, pages 85–94, New York, NY, USA. ACM.

Kohlas, J. and Stärk, R. F. (2007). Information Algebras and Consequence Operators. *Logica Universalis*, 1(1):139–165.

Leduc, G. and Germeau, F. (2000). Verification of security protocols using LOTOS-method and application. *Computer Communications*, 23(12):1089–1103.

Ma, X.-Q. and Cheng, X.-C. (2005). Formal Verification of Merchant Registration Phase of SET Protocol. *International Journal of Automation and Computing*, 2(2):155–162.

Paulson, L. C. (1998). The inductive approach to verifying cryptographic protocols. *Journal of Computer Security*, 6(1–2):85–128.

Sabri, K. E. and Khedri, R. (2006). A multi-view approach for the analysis of cryptographic protocols. In *Workshop on Practice and Theory of IT Security (PTITS 2006)*, pages 21–27, Montreal, QC, Canada.

Sabri, K. E. and Khedri, R. (2007a). A mathematical framework to capture agent explicit knowledge in cryptographic protocols. Technical Report CAS-07-04-RK, department of Computing and Software, Faculty of Engineering, McMaster University. http://www.cas.mcmaster.ca/cas/0template1.php?601 (accessed on May 20, 2009).

Sabri, K. E. and Khedri, R. (2007b). Multi-view framework for the analysis of cryptographic protocols. Technical Report CAS-07-06-RK, department of Computing and Software, Faculty of Engineering, McMaster University. http://www.cas.mcmaster.ca/cas/0template1.php?601 (accessed on May 20, 2009).

Sabri, K. E. and Khedri, R. (2008). Agent explicit knowledge: Survey of the literature and elements of a suitable representation. In *2nd Workshop on Practice and Theory of IT Security (PTITS 2008)*, pages 4–9, Montreal, QC, Canada.

Sabri, K. E., Khedri, R., and Jaskolka, J. (2008). Specification of agent explicit knowledge in cryptographic protocols. In *CESSE 2008 : International Conference on Computer, Electrical, and Systems Science, and Engineering*, volume 35, pages 447–454, Venice, Italy. World Academy of Science, Engineering and Technology.

Sabri, K. E., Khedri, R., and Jaskolka, J. (2009a). Automated verification of information flow in agent-based systems. Technical Report CAS-09-01-RK, department of Computing and Software, Faculty of Engineering, McMaster University. http://www.cas.mcmaster.ca/cas/0template1.php?601 (accessed on May 20, 2009).

Sabri, K. E., Khedri, R., and Jaskolka, J. (2009b). Verification of information flow in agent-based systems. In *E-Technologies: Innovation in an Open World, 4th International Conference, MCETECH 2009*, volume 27 of *Lecture Notes in Business Information Processing*, pages 252–266, Ottawa, Canada. Springer Berlin Heidelberg.

Varadharajan, V. (1990). Petri net based modelling of information flow security requirements. In *Computer Security Foundations Workshop III*, pages 51–61.

Advanced Technologies

Edited by Kankesu Jayanthakumaran

ISBN 978-953-307-009-4
Hard cover, 698 pages
Publisher InTech
Published online 01, October, 2009
Published in print edition October, 2009

This book, edited by the Intech committee, combines several hotly debated topics in science, engineering, medicine, information technology, environment, economics and management, and provides a scholarly contribution to its further development. In view of the topical importance of, and the great emphasis placed by the emerging needs of the changing world, it was decided to have this special book publication comprise thirty six chapters which focus on multi-disciplinary and inter-disciplinary topics. The inter-disciplinary works were limited in their capacity so a more coherent and constructive alternative was needed. Our expectation is that this book will help fill this gap because it has crossed the disciplinary divide to incorporate contributions from scientists and other specialists. The Intech committee hopes that its book chapters, journal articles, and other activities will help increase knowledge across disciplines and around the world. To that end the committee invites readers to contribute ideas on how best this objective could be accomplished.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Khair Eddin Sabri, Ridha Khedri and Jason Jaskolka (2009). Algebraic Model for Agent Explicit Knowledge in Multi-agent Systems, Advanced Technologies, Kankesu Jayanthakumaran (Ed.), ISBN: 978-953-307-009-4, InTech, Available from: http://www.intechopen.com/books/advanced-technologies/algebraic-model-for-agent-explicit-knowledge-in-multi-agent-systems

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821