Small steps in dynamics of information
Velazquez Quesada, F.R.

Citation for published version (APA):
Logical omniscience is not the only idealization Epistemic Logic agents have. In possible worlds models it is taken for granted not only that the agent can recognize as true all the formulas that are so in the worlds she considers possible; it is also assumed that she can talk about any formula. In other words, a possible worlds model assumes that an agent is aware of all formulas of the language. And once again, the idealization leaves out important and interesting actions: this time, changes in awareness.

This chapter focuses on dynamics of the awareness of notion. Such changes are an every-day issue in our life; we can easily imagine situations where new possibilities are introduced (you have lost your keys and someone suggest that you may have left them in the kitchen), and others in which some possibilities are dropped (while watching a soccer match we do not usually think about the finite model property of modal logic).

In order to deal with dynamics of a system, we need the system first. We will start by providing a brief summary of a famous framework for dealing with the awareness of notion: Fagin and Halpern (1988)’s Awareness Logic.

3.1 Awareness Logic

The awareness logic of Fagin and Halpern (1988) is based on two observations. First, the modal operator □ should not be understood as the information the agent actually has, but as the information the agent can eventually get: her implicit information. Second, in order to make explicit her implicit information, the agent should be aware of it.

The awareness logic language extends the base language of EL with an operator A that allows us to build formulas of the form A φ.
Chapter 3. The dynamics of awareness

**Definition 3.1 (Language \( \mathcal{L} \))** Let \( P \) be a set of atomic propositions. Formulas \( \varphi, \psi \) of the awareness language \( \mathcal{L} \) are given by

\[
\varphi ::= p \mid A \varphi \mid \neg \varphi \mid \varphi \land \psi \mid \Box \varphi
\]

with \( p \in P \). Other Boolean connectives \((\lor, \rightarrow, \leftrightarrow)\) as well the existential modal operator \((\Diamond)\) are defined as usual.

In this chapter, formulas of the form \( A \varphi \) are read as “the agent is aware of \( \varphi \)”, and formulas \( \Box \varphi \) as “the agent is informed about \( \varphi \) implicitly”. The language is interpreted in possible worlds models that assign a set of formulas to the agent in each world, representing in this way the information she is aware of.

**Definition 3.2 (Awareness model)** Let \( P \) be a set of atomic propositions. An awareness model is a tuple \( M = \langle W, R, A, V \rangle \) where \( \langle W, R, V \rangle \) is a standard possible worlds model (Definition 1.1), and

- \( A : W \to \varphi(\mathcal{L}) \) is the awareness function, returning the formulas that the agent ‘has in mind’. \( A(w) \) is the agent’s awareness set at \( w \).

As usual, a pointed awareness model \( (M, w) \) also has a distinguished world \( w \).

The semantic interpretation of formulas in \( \mathcal{L} \) is entirely as expected.

**Definition 3.3 (Semantic interpretation)** Let \( (M, w) \) be a pointed awareness model with \( M = \langle W, R, A, V \rangle \). Atomic propositions and boolean connectives are interpreted as usual; for \( A \varphi \) and \( \Box \varphi \) we have:

\[
(M, w) \models A \varphi \text{ iff } \varphi \in A(w) \\
(M, w) \models \Box \varphi \text{ iff } \text{ for all } u \in W, Rwu \text{ implies } (M, u) \models \varphi.
\]

Note how, though the syntax and the semantic representation is the same, the awareness of notion is conceptually different from the access notion we used in the previous chapter. While access sets are understood as ‘what the agent has acknowledged as true’, being aware of is a matter of attention; by saying “the agent is aware of \( \varphi \)” we simply indicate that “the agent entertains \( \varphi \)”.

The concept does not imply any attitude pro or con: the agent may believe \( \varphi \), but also reject it. Stated in other, but related terms, “awareness of” does not imply “awareness that”.

On these models we can impose standard epistemic assumptions about the accessibility relation, such as reflexivity, transitivity, and symmetry. Moreover, further conditions can be imposed on the awareness sets, like closure under commutation for conjunction and disjunction, or being generated by some subset of atomic propositions, according to the specific notion of awareness
3.2. Other options for explicit information

one has in mind. Nevertheless, these requirements are orthogonal to the main concern in this chapter, and we will not assume any of them.

The axiom system for awareness logic is exactly that for the minimal epistemic logic (Table 2.1). Since no special properties about the accessibility relation or the awareness sets are considered, no particular axioms are needed.

Let us turn now to the definition of explicit information. In order for the agent to have explicit information about some formula, besides having it as implicit information, the agent should be aware of it. In other words, the agent needs “to be aware of a concept before [she] can have beliefs about it” (Fagin and Halpern 1988). This yields the following definition for explicit information:

\[ \Box \varphi \land A \varphi \]

**Example 3.1** In the one-world model below, the agent is implicitly informed that \( p \) and also that \( q \). But while she is aware of \( p \), she is not aware of \( q \), so her explicit information about \( p \) and \( q \) differs.

\[
\begin{align*}
(M, w_1) &\models \Box p \land \Box q \\
(M, w_1) &\models A p \land \neg A q \\
(M, w_1) &\models (\Box p \land A p) \land \neg (\Box q \land A q)
\end{align*}
\]

Leaving the rule set function and rule formulas aside, there are three main differences between awareness logic and the framework we presented in the previous chapter. First, A-sets are now interpreted as what the agent is aware of, different from the former “agent’s explicit information”. Second, these sets are allowed to have any formula of the awareness language, without restricting them to the propositional ones like we did. Third, and maybe more interestingly, explicit information is defined now as implicit information plus awareness, \( \Box \varphi \land A \varphi \), different from the \( A \varphi \) we used before.

### 3.2 Other options for explicit information

As we have mentioned before, several authors coincide in that the \( \Box \) operator should not be understood as ‘full-blooded information’ representing what the agent actually has, but as a notion of implicit information, representing what

---

1Such conditions are studied in depth in Fagin and Halpern (1988) and, more recently, in Halpern (2001).
Chapter 3. The dynamics of awareness

she can eventually get. But when it comes to defining the finer notion of explicit information there are different opinions. Even in frameworks similar to the one we have presented in the previous chapter there are variations. Let us leave aside the interpretation of the A-sets for a moment, and review the options.

**Explicit information as a primitive notion** In the previous chapter we assumed a primitive notion of explicit information given by the function A assigning a set of formulas to each possible world. For a proper representation of the knowledge notion, we needed to assume that all formulas in such sets were not only preserved by the accessibility relation but also true in the corresponding world. Since the last property is not preserved by standard model operations, we had to restrict formulas in A-sets to those whose truth-value is not affected by such changes: purely propositional formulas.

**Explicit information as a defined notion** The notion of explicit information can also be defined as a combination of □ and A. Fagin and Halpern (1988) already provides us one candidate, □ϕ ∧ Aϕ, which says that ϕ is explicit information whenever it is implicit information and belongs to the A-set of the evaluation point.

Another interesting option arises when we take a closer look to the consequences of our requirements for dealing with knowledge in the previous chapter. We asked for propositional formulas γ in A-sets to be true (truth: Aγ → γ), for these formulas to be preserved by the accessibility relation (coherence: Aγ → □Aγ) and for this relation to be reflexive (□ϕ → ϕ) (also transitive and symmetric). Note how these properties gives us the following equivalence.

\[ Aγ ↔ □Aγ \quad \text{by coherence (→)} \]  
\[ ↔ □(γ ∧ Aγ) \quad \text{by truth (→) and propositional logic (↔)} \]

In IE_k-models, our definition of explicit information is equivalent to □(γ ∧ Aγ).

What is interesting here is how □(γ ∧ Aγ) encodes the coherence and truth requirements. The formula asks directly for γ to be present in the A-set of all R-accessible worlds and for it to be true in each one of them; hence these two properties are not needed anymore. Now, while coherence was a requirement for our most general class of models, truth and reflexivity were required for dealing with the particular case of true implicit and explicit information, that is, implicit and explicit knowledge. But if we define explicit information as □(ϕ ∧ Aϕ), then just reflexivity is needed in order to have true implicit and explicit information, analogous to what happen in classical Epistemic Logic.

This alternative definition has several advantages. The most important is that since the truth property is no longer necessary, we can lift the restriction of A-sets, allowing them to have any formula of the language, and therefore allowing the agent to have explicit information not only about propositional
facts, but also about her own (and eventually) other agent’s information, and about how this information will change after actions are performed. Having true formulas in A-sets is not needed anymore since, after an action takes place, our new definition automatically ‘recomputes’ what is explicit information. Besides that, we have the validity $\Box (\varphi \land A \varphi) \rightarrow \Box \varphi$, so explicit information is implicit information in the general case.

These two advantages are also shared by Fagin and Halpern (1988)’s definition of explicit information as $\Box (\varphi \land A \varphi)$, but there is another reason that makes our $\Box (\varphi \land A \varphi)$ more appealing. Consider the property of explicit information having implicit positive introspection: if the agent is explicitly informed about $\varphi$, then she has implicit information about this. Under Fagin and Halpern (1988)’s definition of explicit information, this property is expressed by the formula $(\Box \varphi \land A \varphi) \rightarrow \Box A \varphi).$ But this formula is not valid in awareness models, even when we restrict ourselves to those with transitive accessibility relations, a property that characterizes positive introspection in EL. The following model proves it, since at $w_1$ it satisfies $\Box \varphi \land A \varphi$ but not $\Box (\Box \varphi \land A \varphi)$:

![Diagram](attachment:image.png)

Why is this undesirable? Implicit information is understood as “the best the agent can do”. Then, if the agent does not have implicit information about her explicit information, intuitively she will not be able to make explicit this, that is, she will not be able to achieve explicit positive introspection by herself.

With our alternative definition, the notion of implicit positive introspection is expressed by $\Box (\varphi \land A \varphi) \rightarrow \Box A \varphi).$ The formula is not valid in the general class of models, but it is in the class of transitive models. In other words, with our definition, implicit positive introspection depends on the properties of the accessibility relation, just like in classical EL. In the same way, considering an euclidean accessibility relation gives us implicit negative introspection, witness the validity of $\Box (\varphi \land A \varphi) \rightarrow \Box \Box (\varphi \land A \varphi)$.

Finally, recall that in classical EL we have not only the notion of information, $\Box \varphi$, but also the notion of possibility: $\Diamond \varphi$ says that the agent considers $\varphi$ possible. Defining explicit information as $\Box (\varphi \land A \varphi)$ gives us also a very natural notion: $\Diamond (\varphi \land A \varphi)$ says that the agent considers $\varphi$ explicitly possible.

For the mentioned reasons, we will define explicit information as follows:

$$\text{Ex } \varphi := \Box (\varphi \land A \varphi)$$

Once we have fixed a definition of explicit information, it is time to concentrate on our main issue: dynamics of awareness.
3.3 Operations on awareness models

Awareness models suggest a natural and simple dynamics. Though the agent is not logically omniscient, she can get new information by various possibly complex acts. But we want to dig deeper. In line with our definition for explicit information, it also makes sense to look for simple actions transforming models that can be put together to analyze more complex informational acts. We will see later on how these transform explicit information.

Defining the basic actions Our models have two separate components for representing information: the accessibility relation and the awareness sets. The following operations modify these components in a simple way, allowing us to define complex epistemic actions later on.

The consider operation represents an “awareness raising” action:

**Definition 3.4 (The consider operation)** Let $M = \langle W, R, A, V \rangle$ be a model and $\chi$ any formula in $L$. The model $M_{+\chi} = \langle W, R, A', V \rangle$ is $M$ with its awareness sets extended with $\chi$, that is,

$$A'(w) := A(w) \cup \{\chi\} \quad \text{for every } w \in W$$

‘Considering’ extends the formulas that an agent is aware of, but we can also define a drop operation with the opposite effect: reducing awareness sets. This fits with the operational idea that agents can expand and shrink the set of issues having their current attention.

**Definition 3.5 (The drop operation)** Let $M = \langle W, R, A, V \rangle$ be a model and $\chi$ a formula in $L$. The model $M_{-\chi} = \langle W, R, A', V \rangle$ reduces $M$’s awareness sets by removing $\chi$, that is,

$$A'(w) := A(w) \setminus \{\chi\} \quad \text{for every } w \in W$$

This operation can be seen as a form of ‘forgetting’, an action usually disregarded in Dynamic Epistemic Logic (but see van Ditmarsch et al. (2009) and van Ditmarsch and French (2009) for proposals).

The preceding actions affect what an agent is aware of. The next one, known from DEL, modifies her implicit information by discarding those worlds where some observed formula $\chi$ fails:

**Definition 3.6 (The implicit observation operation)** Let $M = \langle W, R, A, V \rangle$ be a model and $\chi$ a formula in $L$. The model $M_{\chi} = \langle W', R', A', V' \rangle$ is given by

- $W' := \{w \in W \mid (M, w) \not\models \chi\}$
- $R' := R \cap (W' \times W')$
- $A'(w) := A(w)$
- $V'(w) := V(w)$. 

and, for every $w \in W'$,
3.3. Operations on awareness models

The observation is implicit because, although it removes worlds, it does not affect what the agent is aware of in the preserved ones.

**Building complex actions** Complex actions can now be built by combining basic ones. As an example, it seems natural to think that a public observation of some fact is in fact done consciously, generating awareness. The corresponding operation of “explicit observation” can be defined in the following way.

**Definition 3.7** The explicit observation operation, analogous in its effect to a public announcement in PAL ([Plaza 1989](#), [Gerbrandy 1999](#)), can be defined by means of an implicit observation followed by an act of consideration:

\[
M_{EO(\chi)} := (M_{\chi!})_{\chi}.
\]

The definition also works if we interchange the order of the operations because we are transforming two independent components of our models.\(^2\)

**Preserving static constraints** Though we have not imposed constraints on the static awareness models, it is interesting to note that some reasonable requirements, like our previous coherence (also called weak introspection on A-sets) or equivalence relations for accessibility, are preserved by our operations.

**Proposition 3.1** Considering preserves coherence and equivalence relations.

Proof. The equivalence property of \( R \) is obviously preserved, since \( R \) is not modified. For coherence, take a world \( w \) in \( M_{\chi!} \) and any \( \varphi \in A'(w) \). Suppose \( Rwu \). If \( \varphi \) is already in \( A(w) \), then \( \varphi \in A(u) \) because \( M \) satisfies the principle, and then \( \varphi \in A'(u) \) by the definition of \( A' \). If \( \varphi \) is not in \( A(w) \), then it should be \( \chi \) itself, which by definition is also in \( A'(u) \).

By a similar argument, the drop operation, too, preserves the two mentioned properties.

**Proposition 3.2** Dropping, too, preserves coherence and equivalence relations.

Finally, our actions of implicit observation have the same effect:

**Proposition 3.3** Implicit observation preserves coherence and equivalence relations.

Proof. Equivalence relations are preserved automatically since we go to a sub-model. Next, for coherence, use the fact that the sub-model \( M_{\chi!} \) has the same awareness sets at its worlds as \( M \), while its epistemic accessibility is a sub-relation of that for \( M \).

It is also worthwhile to notice how some properties one might impose on the A-sets (the truth property, the already mentioned closure under commutation for conjunction and disjunction, or being generated by some subset of atomic propositions, as in [Fagin and Halpern 1988](#)) are not preserved by the operations consider and drop.

\(^2\)Still, one might argue that implicit observation and considering take place simultaneously. While this makes sense, we will not pursue it here.
3.4 The actions in action

Consider the following model:

\[
\begin{array}{c}
p, q \\
\downarrow \\
w_1 \\
\quad \{\} \\
\end{array}
\quad
\begin{array}{c}
p, \neg q \\
\downarrow \\
w_2 \\
\quad \{\} \\
\end{array}
\]

In the leftmost world, \(w_1\), the agent does not have implicit information about \(q\), but she is implicitly informed about \(p\), though not explicitly.

After the agent considers \(p\), we get the model on the right: in both worlds, the agent is now explicitly informed about \(p\).

We do not have the truth requirement of the previous chapter anymore, so our agent can also get explicit information about her own awareness, or implicit and explicit information. Here is how this can happen:

\[
\begin{array}{c}
p, q \\
\downarrow \\
w_1 \\
\quad \{p, \text{Ex} \ p\} \\
\end{array}
\quad
\begin{array}{c}
p, q \\
\downarrow \\
w_2 \\
\quad \{p, \text{Ex} \ p\} \\
\end{array}
\]

When she considers \(\text{Ex} \ p\), we get the model on the left. The agent has explicit information about her having explicit information of \(p\). By acting, she has achieved positive introspection.

Next, consider the above explicit observation of \(q\): an implicit observation followed by consideration of \(q\). This yields the model on the right where \(q\) is now part of the agent’s explicit information.

Finally, dropping \(p\) makes the agent lose earlier explicit information about it (that is, we get \(\neg \text{Ex} \ p\)). Moreover, by our definition of explicit information, she no longer has explicit information that \(\text{Ex} \ p\), since the latter formula is no longer true, and therefore, it is no longer implicit information.\(^3\)

There are many further scenarios with complex many-world patterns, but the above will suffice to show the interest of our setting.

\(^3\)This may seem strange since the formula \(\text{Ex} \ p\) is still in the awareness set of the world, but this only means that the agent is aware of it, not that she still endorses it.
3.5 A complete dynamic logic

In order to express how our dynamic operations affect awareness, implicit and explicit information, we extend the static awareness language with modalities representing each basic operation. If \( \chi \) and \( \varphi \) are formulas in the resulting extended language (still called \( \mathcal{L} \) in this section), then so are

- \( \langle +\chi \rangle \varphi \), there is a way of considering \( \chi \) after which \( \varphi \) is the case.
- \( \langle -\chi \rangle \varphi \), there is a way of dropping \( \chi \) after which \( \varphi \) is the case.
- \( \langle \chi! \rangle \varphi \), there is a way of observing \( \chi \) implicitly after which \( \varphi \) is the case.

**Definition 3.8 (Semantic interpretation)** Let \( (M, w) \) be a pointed awareness model and let \( \chi, \varphi \) be formulas in the extended language \( \mathcal{L} \). Then,

\[
(M, w) \models \langle +\chi \rangle \varphi \iff (M_{+\chi}, w) \models \varphi
\]

\[
(M, w) \models \langle -\chi \rangle \varphi \iff (M_{-\chi}, w) \models \varphi
\]

\[
(M, w) \models \langle \chi! \rangle \varphi \iff (M, w) \models \chi \text{ and } (M_{\chi!}, w) \models \varphi
\]

The universal versions of the modalities are defined as the dual of their respective existential, as usual.

The main difference among the new modalities is the precondition. The agent can consider or drop a formula \( \chi \) without any further requirement, but for her to implicitly observe \( \chi \), \( \chi \) needs to be true. In particular, the lack of precondition and the fact that the operations are functional make the semantic interpretation of the existential and the universal modalities for the consider and the drop operation coincide:

\[
(M, w) \models [+\chi] \varphi \iff (M_{+\chi}, w) \not\models \varphi
\]

\[
(M, w) \models [-\chi] \varphi \iff (M_{-\chi}, w) \not\models \varphi
\]

3.5.1 Dynamic completeness theorem

We now formulate a sound and complete logic for the semantic validities in the extended language \( \mathcal{L} \):

**Theorem 3.1 (Reduction axioms for the action modalities)** The valid formulas of the extended awareness language \( \mathcal{L} \) in awareness models are those provable by the axioms and rules for the static language (Table 2.1; see Section 3.1) plus the reduction axioms and modal inference rules listed in Table 3.1.

These axioms express the syntactic basics of the considering and dropping operations, merged with the axioms of Observation Logic (Section 1.4). For instance, how do the propositions that the agent is aware of change when the agent considers \( \chi \)? Our axioms show the two possibilities. After considering
Chapter 3. The dynamics of awareness

Table 3.1: Axioms and rules for the action modalities.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Axiom</th>
</tr>
</thead>
<tbody>
<tr>
<td>$+_p \vdash (+\chi)p \iff p$</td>
<td>$+_A \vdash (+\chi)A\chi \iff \top$</td>
</tr>
<tr>
<td>$-_- \vdash (-\chi)p \iff p$</td>
<td>$-_- \vdash (-\chi)A\chi \iff A\chi$ for $\varphi \neq \chi$</td>
</tr>
<tr>
<td>$+_v \vdash (+\chi)(\varphi \lor \psi) \iff ((+\chi)\varphi \lor (+\chi)\psi)$</td>
<td>$+_A \vdash (+\chi)(\varphi \lor \psi) \iff ((+\chi)\varphi \lor (+\chi)\psi)$</td>
</tr>
<tr>
<td>$+_o \vdash (+\chi)\Diamond \varphi \iff \Diamond (+\chi)\varphi$</td>
<td>$+_A \vdash (+\chi)\Diamond \varphi \iff \Diamond (+\chi)\varphi$</td>
</tr>
<tr>
<td>$+_N \vdash + \varphi, \text{ infer } [+\chi]\varphi$</td>
<td>$+_N \vdash + \varphi, \text{ infer } [+\chi]\varphi$</td>
</tr>
</tbody>
</table>

$\chi$, the agent is aware of a $\varphi \neq \chi$ if and only if she was aware of $\varphi$ before; but also, considering $\chi$ always makes the agent aware of $\chi$. The drop operation has an analogous effect in the opposite direction. The rest of the axioms are simple commutation clauses, indicating the independence of modifying the domain of worlds and the awareness sets.

3.5.2 How the logic describes our major issues

Our logic states how each basic operator of the language is affected by our three actions. By combining these effects and unfolding the definitions, the logic also explains how the derived notion of explicit information changes under these actions. We discuss a few cases, using our earlier definition $\Box(\varphi \land A\varphi)$, and suppressing detailed calculations:

**Explicit information** For the action of considering $\chi$ and explicit information about a different formula $\varphi$, an application of the reduction axioms gives us the following valid principle

$$[+\chi]\Box \varphi \iff \Box([+\chi]\varphi \land A\varphi) \quad (\text{for } \varphi \neq \chi)$$
The principle states that after considering $\chi$ the agent will be explicitly informed about $\varphi$ if and only if she is already implicitly informed that the considering act will make $\varphi$ true and that she is aware of $\varphi$. One might have expected a simpler direct reduction principle \([+\chi]\Ex \varphi \leftrightarrow \Ex \varphi\), but this formula is not valid in the general case, since the consider action may have changed truth values for sub-formulas of $\varphi$. Nevertheless, for propositional formulas $\gamma$ we do have the validity \([+\chi]\Ex \gamma \leftrightarrow \Ex \gamma\): considering $\chi$ does not affect explicit information about propositional facts different from $\chi$.

In the particular case of explicit information about $\chi$ itself, however, we get the following.

**Fact 3.1** The formula \([+\chi]\Ex \chi \leftrightarrow \Box \chi\) is valid.

*Proof.* Using our reduction axioms, we get
\[
[+\chi]\Ex \chi \leftrightarrow [+\chi]\Box (\chi \land A \chi) \\
\leftrightarrow \Box [+\chi] \chi \land \Box [+\chi] A \chi \\
\leftrightarrow \Box [+\chi] \chi \\
\leftrightarrow \Box \chi
\]

The last step is justified by the following proposition.

**Proposition 3.4** The formula $\chi \leftrightarrow [+\chi] \chi$ is valid.

*Proof.* The reason is that, given our semantics, an act of considering $\chi$ can only change truth values for $A \chi$ and formulas containing it. But then, $\chi$ itself cannot be affected by the operation, since it cannot contain $A \chi$. ■

This shows how a consider action makes implicit information explicit.

Now consider the $K$ axiom, the one to blame for logical omniscience in $EL$:
\[
\Box (\varphi \rightarrow \psi) \rightarrow (\Box \varphi \rightarrow \Box \psi)
\]

This formula is still valid in awareness models, and that is reasonable since implicit information is expected to be closed under logical consequence. But the following formula is also valid
\[
\Ex (\varphi \rightarrow \psi) \rightarrow (\Ex \varphi \rightarrow \Box \psi)
\]

The reason is that, since explicit information is also implicit, $\Ex (\varphi \rightarrow \psi)$ and $\Ex \varphi$ already imply $\Box (\varphi \rightarrow \psi)$ and $\Box \varphi$. Then, considering is the action that ‘fills the gap’, turning explicit the formerly implicit information:
\[
\Ex (\varphi \rightarrow \psi) \rightarrow (\Ex \varphi \rightarrow [+\psi] \Ex \psi)
\]

One might think that the real act here is a richer one of drawing the inference, but in our analysis it is the explicit consideration of the conclusion what ‘gives the last little push’ toward explicit information.\footnote{In Chapter 4 we ask for awareness and acknowledgement of formulas as true in order to get explicit information. Then, the needed acts are those of awareness raising and inference.}
But our proposal can describe more, including the behaviour of explicit information under the drop operation. Here is what happens with formulas $\varphi$ that differ from the dropped $\chi$:

$$\lnot \chi \text{ Ex } \varphi \leftrightarrow \Box (\lnot \chi \varphi \land \varnothing \varphi) \quad \text{(for } \varphi \neq \chi)$$

For explicit information about $\chi$ itself, we get the following.

**Fact 3.2** The formula $\lnot \chi \text{ Ex } \chi \leftrightarrow \Box \bot$ is valid.

**Proof.** Using our reduction axioms as above,

$$\begin{align*}
\lnot \chi \text{ Ex } \chi & \leftrightarrow \lnot \chi \Box (\chi \land \varnothing \chi) \\
& \leftrightarrow \Box (\lnot \chi \chi \land \lnot \chi \varnothing \chi) \\
& \leftrightarrow \Box (\lnot \chi \chi \land \bot) \\
& \leftrightarrow \Box \bot
\end{align*}$$

The validity states that after dropping $\chi$ the agent has explicit information about it if and only if she is implicitly informed about contradictions. In the particular cases of consistent information (technically, seriality for the accessibility relation) or true information (reflexivity), this validity becomes

$$\neg \lnot \chi \text{ Ex } \chi$$

read as “one never has explicit information about $\chi$ after dropping it”.

Still, even after dropping it, the agent does keep $\chi$ as implicit information, witness the following valid law:

**Fact 3.3** The formula $\text{ Ex } \chi \rightarrow \lnot \chi \Box \chi$ is valid.

**Proof.** Again, using the axioms and unfolding the definitions,

$$\begin{align*}
\text{ Ex } \chi & \rightarrow \Box \chi \land \Box \varnothing \chi \\
& \rightarrow \Box \chi \\
& \rightarrow \Box \lnot \chi \chi \\
& \rightarrow \lnot \chi \Box \chi
\end{align*}$$

Our proof uses the following proposition, whose justification is analogous to the one for $\chi \leftrightarrow [+\chi] \chi$ (Proposition 3.4).

**Proposition 3.5** The formula $\chi \leftrightarrow \lnot \chi \chi$ is valid.
Finally, we analyze the effect of an implicit observation over explicit information. For any $\varphi$ and $\chi$, unfolding the definition of explicit information via our axioms (we suppress intermediate steps here) gives
\[
[\chi!] \text{Ex} \varphi \leftrightarrow (\chi \rightarrow \Box ([\chi!] \varphi \land (\chi \rightarrow A \varphi)))
\]
The principle states that after an implicit observation of $\chi$ the agent will be explicitly informed about $\varphi$ if and only if, conditional to the truth of the observation, she is already implicitly informed that this observing act will make $\varphi$ true and that she is aware of $\varphi$. This outcome is our solution to the earlier-mentioned problem of update making explicit information ‘out of synch’ with reality. (Recall that this was the reason for the restriction to purely factual assertions in the previous chapter.) Explicit information is now a defined notion, so it automatically re-adjusts to whatever happens to the modalities $\Box$ and $A$, and our logic tells us precisely how.

We have extracted the effect of our basic epistemic actions over explicit information defined as $\Box (\varphi \land A \varphi)$. Thus, we replace discussion whether the agents’s information is closed under logical consequence by a much richer picture of what they can do to change their information.

Moreover, this style of analysis works not only for the stated notion of explicit information; it can also provide us with validities expressing the way different definitions of explicit information are affected by dynamic actions, like Fagin and Halpern (1988)’s $\Box \varphi \land A \varphi$ or others, like $\Box \varphi \land A \Box \varphi$.

### 3.5.3 Schematic validities and algebra of actions

While all this seems a straightforward dynamic epistemic technique, there is a catch. In deriving the principles of the previous section, we have used more than the reduction axioms of our logic per se. Several important ‘schematic’ principles did not follow from our reduction axioms. In particular, we have used the two principles
\[
[+\chi] \chi \leftrightarrow \chi \quad \text{and} \quad [-\chi] \chi \leftrightarrow \chi
\]
whose validity involved additional considerations. Of course, each specific instance of such a formula can be derived, given our completeness theorem. But that does not mean there is any illuminating uniform derivation of an “algebraic” sort. Indeed, an explicit characterization of the schematic validities in dynamic-epistemic logics (valid for all substitutions of formulas for proposition letters) is a well-known open problem (cf. van Benthem (2010)), even in the case of Public Announcement Logic. Given the importance of such general principles here, that problem becomes even more urgent.
Algebra of actions We end this section with one particular source of schematic validities. As important as it is to understand how actions affect our information, their general algebraic structure is of interest too. We briefly discuss some validities, to show that this “algebra of actions” raises some interesting issues:

- In general, drop does not neutralize consider: 
\[ [+\chi] [-\chi] \varphi \leftrightarrow \varphi \] is not valid. If the agent is initially aware of \( \chi \), consider makes no change, but drop does, yielding a model where \( \chi \) is not in the awareness set. The actual validity is the qualified
\[ \neg A\chi \rightarrow ([+\chi] [-\chi] \varphi \leftrightarrow \varphi) \]

- The dual case behaves in the same way: consider does not neutralize drop in general, but we do have:
\[ A\chi \rightarrow ([-\chi] [+\chi] \varphi \leftrightarrow \varphi) \]

As for unqualified algebraic laws, we do have idempotence:

- A sequence of consider actions for the same formula has the same effect as a single one, and the drop operation behaves similarly:
\[ [+\chi] \varphi \leftrightarrow [+\chi] [+\chi] \varphi \quad \text{and} \quad [-\chi] \varphi \leftrightarrow [-\chi] [-\chi] \varphi \]

Next, given the dynamics of the system, we do not expect strong commutation laws between considering and dropping (the fact that they do not cancel each other gives us a clue). Nevertheless, we do expect commutation of these operations with implicit observation, since the latter modifies an independent component of our models. For example, the following formulas
\[ [\chi!] [+\chi] \varphi \leftrightarrow [+\chi] [\chi!] \varphi \quad \text{and} \quad [\chi!] [-\chi] \varphi \leftrightarrow [-\chi] [\chi!] \varphi, \]
are valid even for formulas \( \chi \) using the modality \( A \). The reason is, once again, that the operations \( +\chi \) and \( -\chi \) can only change the truth value of \( A\chi \), and hence that of \( \chi \) cannot be affected.

This action algebra, yielding more validities when we restrict our attention to just factual assertions, clearly involves uniform schematic validities that once more are not immediately obvious from our earlier completeness theorem. In fact, \( PAL \) itself (what we have called observation logic) has an algebra of actions describing the behaviour of successive announcements, but it tends to go unnoticed since two successive announcements can be compressed into a single one, as the following validity indicates:
\[ [\chi_1!] [\chi_2!] \varphi \leftrightarrow ([\chi_1] \land [\chi_1!] [\chi_2!] \varphi \]

This compression disappears when the operation changes the accessibility relation, as it is done in dynamic epistemic logics for changes in preferences or beliefs: two successive upgrades cannot be compressed into a single one (van Benthem and Liu 2007; van Benthem 2007).
3.6 From single to multi-agent scenarios

So far, we have considered activities of single agents, including not only their observations, but also their acts of awareness raising. Now, the latter are typically private, and hence it makes sense to look at scenarios with privacy. But a bit paradoxically, privacy only becomes visible in a multi-agent setting. Here is a first simple illustration with two agents:

Example 3.2 Consider the following model $M$, generalizing the single-agent framework to a multi-agent setting in a straightforward way:

In the unique world of the model, $w_1$, each agent is implicitly informed about $p$, but no agent is aware of $p (\neg A_1 p \land \neg A_2 p)$. Moreover, agents have implicit information about each other’s lack of awareness about $p$. E.g., agent 2 is implicitly informed that agent 1 is not aware of $p (\Box_2 \neg A_1 p)$.

Now let an event take place: agent 1 considers $p$. The model $M_{+p_1}$ is given by

In the new situation, agent 1 is aware of $p (A_1 p)$, and now has explicit information about it. But there is more: though agent 2 does not have any new explicit information, she is now implicitly informed that agent 1 is aware of $p (\Box_2 A_1 p)$.

Is this a realistic scenario? Independently of the modelling, it seems strange that an internal action that takes place only in agent 1’s mind can affect immediately the information of agent 2. This shows the need of a more detailed analysis of how awareness models should change in a setting that allows not only public but also private actions.

3.6.1 Multi-agent static framework

The extension of the static awareness framework to a setting with many agents in a group $Ag$ is straightforward. In the language of multi-agent $\mathcal{L}$, we just add agent indexes to the $A$ and the $\Box$ modalities ($A_i$ and $\Box_i$, respectively). In the semantic models, $R$ becomes a function from $Ag$ to $\wp(W \times W)$ returning an accessibility relation $R_i$ for each agent $i \in Ag$, and $A$ becomes a function from $Ag \times W$ to $\wp(\wp)$ returning the awareness set $A_i(w)$ of each agent $i$ at each possible world $w$. The semantic interpretation of formulas is then as before, using $A_i$ and $R_i$ to interpret formulas of the form $A_i \varphi$ and $\Box_i \varphi$, respectively.
Again, in this multi-agent case we will not impose special semantic con-
straints. But it is interesting to notice that if we had been dealing with the
notion of knowledge, going from public to private actions would have made us
to change the notion to one in which information is not required to be true (just
as in DEL in general). This is because, being unable to observe some actions,
an agent’s information can go out of synch with reality.

3.6.2 Multi-agent actions: the general case

To make our actions private, we need a mechanism in which we can represent
actions that affect different agents in different ways. The action models of Baltag
et al. (1999) allow us to do that. The key observation behind them is that, just
as the agent can be uncertain about which one is the real world, she can also
be uncertain about which particular event has taken place. In such situations,
the uncertainty of the agent about the action can be represented with a model
similar to that used for representing her uncertainty about the static situation.

More precisely, an action model consists of a collection of possible events
connected by means of an accessibility relation, indicating the events each agent
considers possible. Different from the worlds of a possible worlds model, and
as their name indicate, each event is not understood as a possible state of affairs,
but as an event that might have taken place. Instead of associating them with
an atomic valuation, each event is associated with a precondition, indicating
what that particular event requires to take place.

In order to make action models suitable for our purposes, we will extend
them in essentially the manner of van Benthem et al. (2006) where, besides
affecting the agent’s uncertainty, action models can also affect the real world.
In our case, besides affecting the agent’s uncertainty, our action models will
also be able to affect the agent’s awareness.

Definition 3.9 (Multi-agent action model) With \( P \) the set of atomic proposi-
tions and \( Ag \) the finite set of agents, a multi-agent action model is a tuple
\( C = \langle E, T, \text{Pre}, \text{Pos}_A \rangle \) where

- \( \langle E, T, \text{Pre} \rangle \) is an action model (Baltag et al. 1999) with \( E \) a finite non-empty
  set of events, \( T : Ag \rightarrow \wp(W \times W) \) a function returning an accessibility
  relation \( T_i \) for each agent \( i \in Ag \) and \( \text{Pre} : E \rightarrow L \) the precondition function
  indicating the requirement for each event to be executed;

- \( \text{Pos}_A : (Ag \times E \times \wp(L)) \rightarrow \wp(L) \) is the postcondition function, assigning a
  new set of formulas in \( L \) to every tuple of an agent, event, and (old) set of formulas in \( L \).

A pointed action model \( (C, e) \) has a distinguished event \( e \).
Observe how, in our action model, each event \( e \) comes not only with a precondition \( \text{Pre}(e) \), a formula expressing what \( e \) needs to take place, but also with a postcondition \( \text{Pos}_{A_i}(e, X) \), the set of formulas agent \( i \) would be aware of if \( e \) takes place. This function \( \text{Pos}_{A_i} \) is a generalization of the substitution function in \[\text{van Benthem et al.}(2006)\] for representing factual change.

We have defined the way we will represent our actions. It is now time to define how they will affect the agent’s information.

**Definition 3.10 (Product update)** Let \( M = \langle W, R, A, V \rangle \) be a multi-agent awareness model and let \( C = \langle E, T, \text{Pre}, \text{Pos}_{A}\rangle \) be a multi-agent action model. The **product update** operation \( \otimes \) yields the model \( M \otimes C = \langle W', R', A', V' \rangle \), given by

- \( W' := \left\{ (w, e) \mid (M, w) \not\models \text{Pre}(e) \right\} \)
- \( R'_i(e_1)(w_2, e_2) \iff R_i(w_1w_2) \text{ and } T_i e_1 e_2 \)

and, for every \((w, e) \in W'\),
- \( V'(w, e) := V(w) \)
- \( A'_i(w, e) := \text{Pos}_{A_i}(e, A_i(w)) \)

The set of worlds of the model \( M \otimes C \) is given by the restricted cartesian product of \( W \) and \( E \): a pair \((w, e)\) will be a world in the new model if and only if event \( e \) can be executed at world \( w \). For each agent \( i \), her uncertainty about the situation *after* an action, \( R'_i \), is a combination of her uncertainty about the situation *before* the action, \( R_i \), and her uncertainty about the action, \( T_i \). The agent will not distinguish \((w_2, e_2)\) from \((w_1, e_1)\) if and only if she does not distinguish \( w_2 \) from \( w_1 \) and \( e_2 \) from \( e_1 \). Or, from the opposite perspective, the agent will distinguish \((w_2, e_2)\) from \((w_1, e_1)\) if and only if she already distinguishes \( w_2 \) from \( w_1 \), or \( e_2 \) from \( e_1 \). For the new atomic valuation, each world \((w, e)\) inherits that of its static component \( w \): an atom \( p \) holds at \((w, e)\) if and only if \( p \) holds at \( w \).

Now observe how the function \( \text{Pos}_{A_i} \) works: for each agent \( i \) and each event \( e \), \( \text{Pos}_{A_i} \) takes agent \( i \)'s awareness set at \( w \) in \( M \), and returns her awareness set at \((w, e)\) in \( M \otimes C \). Note how \( \text{Pos}_{A_i} \) does not have restrictions on the format of the definition; in fact, it can even return a new awareness set that is completely unrelated to the original one. The cases of interest in this chapter have a simple definition, and more uniform expressions will be explored in Chapter \[\text{v}\].

In order to express how product updates affect the agents’ information, the *extended* multi-agent language \( \mathcal{L} \) has extra modalities: if \((C, e)\) is a pointed action model and \( \varphi \) is a formula in the extended multi-agent \( \mathcal{L} \), then so is \( \langle C, e \rangle \varphi \). The semantic interpretation of these new formulas is as follows:

**Definition 3.11 (Semantic interpretation)** Let \((M, w)\) be a pointed multi-agent model and let \((C, e)\) be a pointed action model with \( C = \langle E, T, \text{Pre}, \text{Pos}_{A}\rangle \).

\[
(M, w) \models \langle C, e \rangle \varphi \iff (M, w) \not\models \text{Pre}(e) \text{ and } (M \otimes C, (w, e)) \models \varphi
\]

It is time to look at concrete cases illustrating the mechanism.
3.6.3 Public consider and drop

To begin with, our multi-agent setting generalizes the single agent case, since we can define action models for our earlier (now public) consider and drop operations.

**Definition 3.12 (Public consider action)** Let $\chi$ be a formula in multi-agent $\mathcal{L}$. The action of agent $j$ publicly considering $\chi$ is given by the pointed action model $(\text{Pub}_+^j, e_1)$ with $\text{Pub}_+^j = \langle E, T, \text{Pre}, \text{Pos}_A \rangle$ defined as

\[
\begin{align*}
E & := \{e_1\} \\
T_i & := \{(e_1, e_1)\} \text{ for every agent } i \\
\text{Pre}(e_1) & := \top \\
\text{Pos}_{A_j}(e_1, X) & := X \cup \{\chi\} \\
\text{Pos}_{A_i}(e_1, X) & := X \text{ for } i \neq j
\end{align*}
\]

The diagram on the left shows the action model $\text{Pub}_+^1$ in the 2-agent case (with the precondition omitted).

**Definition 3.13 (Public drop action)** Let $\chi$ be a formula in multi-agent $\mathcal{L}$. The action of agent $j$ publicly dropping $\chi$ is given by the pointed action model $(\text{Pub}_-^j, e_1)$, which differs from a public considering only in its postcondition function for $j$ in $e_1$:

\[
\text{Pos}_{A_j}(e_1, X) := X \setminus \{\chi\}
\]

The public versions of the actions have just one event, and their accessibility relations $T_i$ indicate that all involved agents recognize this. Moreover, the precondition in the unique world is simply $\top$. Then, the application of $(\text{Pub}_+^j, e_1)$ ($(\text{Pub}_-^j, e_1)$, respectively) on a multi-agent static model $M$ yields a copy of $M$ in which $\chi$ has been added to (removed from, respectively) the awareness set of agent $j$ in all worlds.

3.6.4 Private consider and drop

But our mechanism can also define private actions. Here are simple versions of the earlier consider and drop. As usual, these encode what takes place, but also how different agents ‘view’ this.
Definition 3.14 (Private consider action) Let $\chi$ be a formula in multi-agent $\mathcal{L}$. The action of agent $j$ privately considering $\chi$ is given by the pointed action model $(\text{Pri}_{+\chi}^j, e_1)$ with $\text{Pri}_{+\chi}^j = \langle E, T, \text{Pre}, \text{Pos}_A \rangle$ defined as

- $E := \{e_1, e_2\}$
- $T_i := \begin{cases} \{(e_1, e_1), (e_2, e_2)\} & \text{if } i = j \\ \{(e_1, e_2), (e_2, e_2)\} & \text{otherwise} \end{cases}$
- $\text{Pre}(e_1) = \text{Pre}(e_2) := \top$
- $\text{Pos}_A^j(e_1, X) := X \cup \{\chi\}$, $\text{Pos}_A^j(e_2, X) := X$
- $\text{Pos}_A^i(e_1, X) := X$, $\text{Pos}_A^i(e_2, X) := X$ for $i \neq j$

The diagram on the left shows the model $\text{Pri}_{+\chi}^1$ for 2 agents (preconditions again omitted).

Definition 3.15 (Private drop action) Let $\chi$ be a formula in multi-agent $\mathcal{L}$. The action of agent $j$ privately dropping $\chi$ is given by the pointed action model $(\text{Pri}_{-\chi}^j, e_1)$, which differs from a private considering only in its postcondition function for $j$ in $e_1$:

- $\text{Pos}_A^j(e_1, X) := X \setminus \{\chi\}$

The difference between the public and the private version of the actions is that the private actions involve two events: one in which $\chi$ is added to (removed from) agent $j$’s awareness set (the event $e_1$), and another in which there is no change (the event $e_2$). Moreover, the accessibility relations $T_i$ indicate that, while $j$ recognizes which event is the real one (our $e_1$), the other agents do not consider that event possible, sticking to the ‘no change’ option. Then, the application of $(\text{Pri}_{+\chi}^j, e_1)$ $(\text{Pri}_{-\chi}^j, e_1)$, respectively on a multi-agent static model $M$ yields a model containing two copies of $M$: one, recognized as the real one only by $j$, in which $j$’s awareness set has changed, and another, viewed by the other agents as the only possibility, in which nothing has happened.

Example 3.3 Recall the model $M$ from Example 3.2. After agent 1 considers $p$ privately (i.e., after applying $(\text{Pri}_{+p}^1, e_1)$), we get a better version of the initial situation that started the thread of this section:
3.6.5 Unconscious versions

The flexibility of the postcondition mechanism is great. We can represent many further scenarios, even unconscious actions, hidden from all agents, including the one that ‘performs’ it! We just give an illustration:

**Definition 3.16 (Unconscious drop action)** Let $\chi$ be a formula in the multi-agent $\mathcal{L}$. The action of agent $j$ unconsciously dropping $\chi$ is given by the pointed action model $(\text{Unc}^j_{\chi}, e_1)$, differing from its private counterpart only in the definition of the accessibility relation:

- $T_i := \{(e_1, e_2), (e_2, e_2)\}$ for all agents $i$

The diagram on the left depicts $\text{Unc}^1_{\neg \chi}$ in a 2-agent scenario.

**Example 3.4** Consider the model $(M \otimes \text{Pri}^{1}_{(p, e_1)})$ of Example 3.3 If agent 1 unconsciously drops $p$, we get the following updated model:

In the evaluation point, $(w_1, e_1)$, agent 1 is aware of $p$ ($A_1 p$), just like she does after publicly considering $p$. But this time, agent 2’s implicit information does not change: she is still implicitly informed that agent 1 is not aware of $p$ ($\Box_2 \neg A_1 p$).
3.6. From single to multi-agent scenarios

The updated model contains two copies of the original one. Here, agent 1 considers only the rightmost world of the upper copy, and agent 2 only the rightmost world of the lower one.

Much more can be said about this scenario, and we feel that we have a promising take here on unconscious actions such as forgetting. But our purpose here was just to demonstrate the flexibility of the framework.

3.6.6 Completeness of the multi-agent system

In principle, there is a complete axiom system for our product update mechanism for awareness, and it looks like our earlier single-agent logic, with indices attached. Its principles for atomic formulas, boolean operations, and implicit knowledge are the usual ones from Dynamic Epistemic Logic DEL. As an illustration, we have the valid equivalence

\[ \vdash \langle C, e \rangle \Diamond_i \varphi \leftrightarrow \text{Pre}(e) \rightarrow \bigvee_{T, e, f} \Diamond_i \langle C, f \rangle \varphi \]

But to formulate a precise result, the crucial issue is stating the right reduction axiom for awareness given the postconditions. Consider the earlier axioms that we gave for our two basic syntactic operations of consider and drop. These described the postconditions (the effect of the operations on the A-sets) inside the language, exploiting the simple format of their effects. For instance, we have \( \varphi \) in our A-set after an act \( +\chi \) if we had \( \varphi \) before, or \( \varphi \) is actually the just added formula \( \chi \). This case distinction in the reduction axiom reflects directly the simple disjunctive definition of the postcondition for the action \( \chi!: A'(w) := A(w) \cup \{\chi\} \).

The same is true in our more general setting with action models: simple definitions of postconditions in our action models will allow us to provide matching reduction axioms. Consider, as an illustration, the system in van Benthem et al. (2006); it allows factual change by modifying the set of worlds in which each atomic proposition is true. But this new set is not an arbitrary one; it is defined syntactically by means of a formula of the language. This suggest that, in order to get a proper completeness theorem, we should look for some uniform syntactic expressions from which reduction axioms can be derived. We will deal with this issue in Chapter 5.

Though we have merely made some proposals, the defined actions show the power of a simple syntactic extension of the well-known DEL action models and its product update.
Chapter 3. The dynamics of awareness

3.7 Remarks

In this chapter we have noticed that, besides the lack of acknowledgement of a given formula as true, there is another reason for which an agent may have just implicit information about it: she may not be aware of it. This notion has been the main protagonist of the present chapter. Based in the Awareness Logic of [Pagin and Halpern (1988)], we have discussed its role in the definition of the notion of explicit information, presenting several possibilities that make also use of the implicit information notion. Table 3.2 summarizes the notions of information discussed in this chapter and the definition we have worked with.

<table>
<thead>
<tr>
<th>Notion</th>
<th>Definition</th>
<th>Relevant model requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Awareness of $\mathsf{A} \varphi$</td>
<td>——</td>
<td></td>
</tr>
<tr>
<td>Implicit information $\Box \varphi$</td>
<td>——</td>
<td></td>
</tr>
<tr>
<td>Explicit information $\Box (\varphi \land \mathsf{A} \varphi)$</td>
<td>——</td>
<td></td>
</tr>
</tbody>
</table>

Table 3.2: Static notions of information.

Here it is important to emphasize again the difference in the interpretation of the $\mathsf{A}$-sets in this and the previous chapter. In Chapter 2 we discuss an agent that does not need to be omniscient because she does not need to recognize every true formula as such; in that case the $\mathsf{A}$-sets are interpreted directly as the agent’s explicit information. In the present chapter we have discussed an agent that does not need to be omniscient because she does not need to have full attention; in this case the $\mathsf{A}$-sets are interpreted as those formulas the agent is aware of, but this does not imply any attitude, positive or negative, about them. Note that awareness by itself is not enough to provide the agent explicit information (we are aware of many possibilities, but that definitely does not imply that we assume that all of them are true); this is another reason why we have changed our definition of explicit information from the $\mathsf{A} \varphi$ of the previous chapter, to the current $\Box (\varphi \land \mathsf{A} \varphi)$.

On the dynamic side, just like in the previous chapter, the introduction of finer notions of information has allowed us to describe acts that, though important for real human agents, have been neglected in the classical DEL literature due to the strong idealization of the represented agents. In this case, the highlighted acts are those that change the information the agent is aware of (therefore modifying her explicit information), and we have provided a formal representation for two of them: consider, an act of awareness raising, and drop, an act of awareness reduction. For the third static notion discussed in this chapter, implicit information, we have not only recalled an action that affects it, implicit observation (the PAL public announcement), but also shown how an
explicit version, *explicit observation*, can be defined with the help of the consider action. Moreover, we have made the jump from a single-agent case to a multi-agent scenario, and we have observed that, in public, privacy becomes important. Accordingly, we have provided proper multi-agent representation for private and even unconscious versions of the awareness-changing acts already proposed for the single-agent case. These defined actions sketch the power of a syntactic extension of well-known action models and product update, and further examples of their application will be presented in Chapter 5. Table 3.3 summarizes the actions that have been defined in this chapter.

<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Consider (in its public and private versions).</td>
<td>The agent becomes aware of a formula.</td>
</tr>
<tr>
<td>Drop (in its public and private versions).</td>
<td>The agent loses awareness of a formula.</td>
</tr>
<tr>
<td>Implicit observation.</td>
<td>Changes the agent’s implicit information.</td>
</tr>
<tr>
<td>Explicit observation.</td>
<td>Changes the agent’s implicit and explicit information.</td>
</tr>
</tbody>
</table>

Table 3.3: Actions and their effects.

The notion of awareness (and its dual, unawareness) has been an interesting research topic not only in Logic (Ågotnes and Alechina 2007; van Ditmarsch et al. 2009) but also in Computer Science Halpern (2001); Halpern and Régó (2005, 2009) and particularly in Economics (Modica and Rustichini 1994; Dekel et al. 1998; Modica and Rustichini 1999; Heifetz et al. 2003; Board and Chung 2006; Sillari 2006; Samet 2007). Dynamics of the notion have been recently explored also in van Ditmarsch and French (2009) and Hill (2010). Our particular approach shows how a significant informational dynamics can take place over existing awareness models, generalizing acts of observation and awareness change. It also shows how this leads to useful technical systems, and we have provided results about them in the spirit of Dynamic Epistemic Logic. Thus, we have shown that the ‘reductionist approach’ to explicit knowledge in terms of implicit semantic knowledge and syntactic awareness is feasible and interesting in its own right.

During the present chapter some new issues have arisen. Our multi-agent setting can describe many more agent activities than what we have shown, and we have only scratched the surface. Also, many technical issues remain open, like the issue of schematic validities and action algebra. In the case of the first, there is already one interesting result: the set of schematic validities in *Public Announcement Logic* is decidable (Holliday et al. 2010).
But beyond this, there is a more important question: how the two notions of explicit information that we have worked with so far are related? In Chapter 2 we worked under the intuition that the only reason why implicit information may not be explicit is because the agent could fail to recognize true formulas as such, just like we may fail to recognize that the conclusion of a theorem is true. But it is now clear that acknowledging a formula as true could not be enough because we could still need the adequate ‘attention’ to the subject.

In the present chapter we have reduced explicit information to implicit information plus awareness, that is, the only reason why implicit information may not be explicit is because the agent could not have full attention, just like we may fail to recognize that our lost keys are in the kitchen because we do not even consider that possibility. But it is also clear that full attention could not be enough to make explicit our implicit information because we could still need to recognize the information as true. Think of a conclusion that I am pondering, and that in fact follows from some premises whose truth I explicitly have. I could still fail to see explicitly the conclusion. In fact, this shows how our acts of awareness raising are not acts of inference, since under this chapter’s definition, merely becoming aware of \( \varphi \) was enough to upgrade information from implicit to explicit.

A more satisfying notion of explicit information is one that combines the two ideas: the agent needs to be aware of the subject, but also recognize it as true. This idea, and the focus on the particular case of true information (knowledge) will be the topic of our next chapter.