Table 22.2 The Fit between UP and AM

PRACTICE	FIT
Active Stakeholder Participation	AM has a wide definition for project stakeholders, including users, management, operations staff, and support staff to name a few. This definition is compatible with the UP. The UP clearly includes project stakeholders, such as users and customers, throughout most of its disciplines. To be successful, UP project teams should allow project stakeholders to take on modeling roles such as Business Process Designer and Requirements Specifier as appropriate; there is nothing in the UP that prevents this. The more active project stakeholders are, the less of a need there will be for reviews, management presentations, and other overhead activities that reduce your team's development velocity.
Apply Modeling Standards	The application of modeling standards, in particular the diagrams of the UML, is a significant part of the UP. Furthermore, the RUP product (Rational Corporation 2001) includes guidelines for the creation of many modeling artifacts, guidelines that your teams should consider adopting and following as appropriate, and explicitly suggests that you tailor the guidelines that they provide for your exact needs. To remain agile, however, UP teams must recognize that you often need to bend the guidelines and standards. In other words, don't let them become a straight jacket.
Apply Patterns Gently	UP teams are free to apply modeling patterns. The RUP product describes many common modeling patterns as part of their efforts for a variety of modeling disciplines. This practice enhances the UP with its advice to ease into the application of a pattern. The UP does not make this concept as explicit as it could.
Apply The Right Artifact(s)	One of the strengths of the UP is that it provides some advice for when to create each type of model. Recent incarnations of the RUP product include significant advice for non-UML artifacts such as data models and user interface storyboards (UI flow diagrams).
Collective Ownership	AM's concept of collective ownership can be used to enhance the efforts on UP projects, assuming that the team culture supports the concept of open and honest communication. The UP supports collective ownership with its strong focus on configuration management issues (it has a discipline dedicated to this task), although its change management processes may potentially get in your way if developers and project stakeholders are unable to distinguish when to formalize change control and when not to. To be fair, this is a problem regardless of when you apply AM on an UP project, or on any type of project for that matter. UP teams should turn the configuration management dial up a few notches and allow anyone on the project to access and work on any artifact that they wish, including models and documents.
Consider Testability	The UP includes a Test discipline in its lifecycle, making testing an explicit issue that everyone should keep in mind as they work. The UP also includes many opportunities to review modeling artifacts, if you choose to follow this form of validation. To fully adopt this practice the consideration "Is it testable" should be included in all modeling activities.

Table 22.2 continued

PRACTICE	FIT
Create Several Models in Parallel	The UP clearly includes this concept. One only has to look at the activity diagrams that depict each discipline to see that several artifacts are potentially worked on in parallel. However, this concept could be communicated better because the near-serial flow in the activity diagrams presented for each major modeling activity doesn't communicate this concept well. There is a larger issue as well when you consider the lifecycle as a whole. Because the UP has organized its modeling efforts into separate disciplines, for very good reasons, it isn't as apparent that not only can you work on several business modeling artifacts in parallel but you can also work on requirements-oriented artifacts, analysis-oriented artifacts, architecture artifacts, and design artifacts too. UP teams can turn the dial up a few notches by reading between the lines of the discipline activity diagrams and the UP lifecycle diagram and choosing to perform activities from several disciplines simultaneously when it makes sense to do so.
Create Simple Content	This practice is a choice made by the modeler(s), albeit one that must be implicitly supported by the rest of the development team. UP teams will need to adopt modeling guidelines that allow models that are just good enough and the customers of those models (including programmers, project stakeholders, and reviewers) must also be willing to accept simple models. This is a cultural issue, one that is often difficult for many organizations to adopt.
Depict Models Simply	See *Create Simple Content.*
Discard Temporary Models	Modelers on UP teams are free to discard anything that they wish. As with the Simplicity practices your organization's culture must accept the concept of traveling light, of developing and maintaining just enough models and documents, and no more.
Display Models Publicly	UP teams are free to follow this practice. UP teams can turn the communication dial up a notch by following the principle *Open and Honest Communication* by making all artifacts available to everyone as well as to publicly display the critical models used by the project team.
Formalize Contract Models	The UP includes the concept of integrating with external systems. These systems are typically identified on use case models and the RUP suggests introducing "boundary classes" to implement the interface to these systems. At the time of this writing the RUP appears weak with respect to activities such as legacy system analysis and enterprise application integration (EAI). The explicit adoption of this practice clearly strengthens the UP's integration activities and fits in well with it's concepts of use case realizations—the interaction between systems can be specified with one or more use cases and then the corresponding use case realization becomes the formalized contract model.

continues

Table 22.2 The Fit between UP and AM (continued)

PRACTICE	FIT
Iterate To Another Artifact	This practice can be easily adopted by UP teams. As mentioned previously, the unfortunate depiction of UP modeling activities as quasi-serial processes and the division of modeling activities into separate disciplines can hinder the iterative mindset required of agile modelers.
Model in Small Increments	This practice is clearly an aspect of the UP. The UP's support for iterations implies that you can incrementally develop your model throughout the project. UP teams can easily turn the iterative and incremental dial up a few notches by preferring smaller, simpler models that quickly lead to implementation and testing.
Model to Communicate	The UP implicitly includes this practice. UP teams can turn the communication dial up a few notches by following the principle *Model With a Purpose* by knowing who their audience for the model is and what they require of that model.
Model to Understand	See *Model to Communicate*.
Model With Others	The UP implicitly includes this practice. Every modeling discipline clearly includes several roles; each role is filled by one or more people. UP teams can turn the communication dial up a few notches by adopting tools that support team modeling, such as whiteboards and collaborative modeling tools (see Chapter 8) over single-user modeling tools.
Prove it With Code	The UP explicitly includes this practice. At the end of each "iteration, except perhaps for the ones during the Inception phase (Ambler and Constantine 2000a), the UP specifically states that you should have a working prototype. Furthermore, the UP insists that you have a working end-to-end prototype at the end of the Elaboration phase (Ambler and Constantine 2000b) that proves your architecture.
Reuse Existing Resources	Reuse is an implicit part of the UP, and reuse management is an explicit part of the EUP. UP teams can turn the reuse dial up a few notches by actively preferring to reuse existing resources instead of building them from scratch, including but not limited to existing models, existing components, open source software (OSS), and existing tools.
Update Only When It Hurts	In theory this can be an easy concept for UP teams to adopt as it dramatically reduces the effort expended to keep your artifacts up to date. However, in practice many organizations prove to have a problem with this concept, particularly if they have a strong "traceability" culture. Traceability is the ability to relate aspects of project artifacts to one another, the support for which is a strong feature of the UP because it's an important aspect of its Configuration and Change Management discipline. Furthermore, the RUP product includes tool mentors for working with Rational RequisitePro (hyperlink "http://www.rational.com"), a requirements traceability tool, making it appear easy to

Table 22.2 continued

PRACTICE	FIT
	maintain a traceability matrix between artifacts. My experience is that organizations with traceability cultures will often choose to update artifacts regularly, even if it isn't yet painful to have the artifacts out of date, and update the traceability matrix that relates everything to one another. To turn their productivity dial up several notches UP teams should choose to travel light, to loosen up a bit and allow project artifacts to get out of sync with one another. Maintain a traceability matrix between artifacts only when there is a clear benefit to do so AND their project stakeholders understand the issues involved as well as authorize the effort. A traceability matrix is effectively a document and is therefore a business decision made by project stakeholders.
Use the Simplest Tools	The RUP product includes tool mentors that make it easier for teams to work with tools sold by Rational Corporation. However, the reality is that UP teams are welcome to use any development tool that they want and Rational tools compete on their merits just like the products of any other company. UP teams can turn their productivity dial up several notches by expanding their horizons to include simple tools such as whiteboards, index cards, and sticky notes in addition to CASE tools.

Choose To Be Agile

In this chapter you've seen that modeling is an important part of the Unified Process, the RUP instantiation of it includes three disciplines: Business Modeling, Requirements, and Analysis and Design. EUP adds a fourth discipline, Infrastructure Management that includes significant modeling efforts. You've also seen that the potential exists to take an UP/AM approach to development, if you choose to do so. The next chapter explores how AM can be used to enhance the UP disciplines and Chapters 23 through 27 work through examples of taking an UP/AM approach to the SWA Online case study. Chapter 28 explores specific issues that pertain to adopting AM on a UP project. The greatest issue that UP teams face with respect to AM is the need to take an agile approach to development, which is possible for an UP project team to do, but in practice proves to be a difficult path to take. Choosing to succeed is often the most difficult decision to make.