

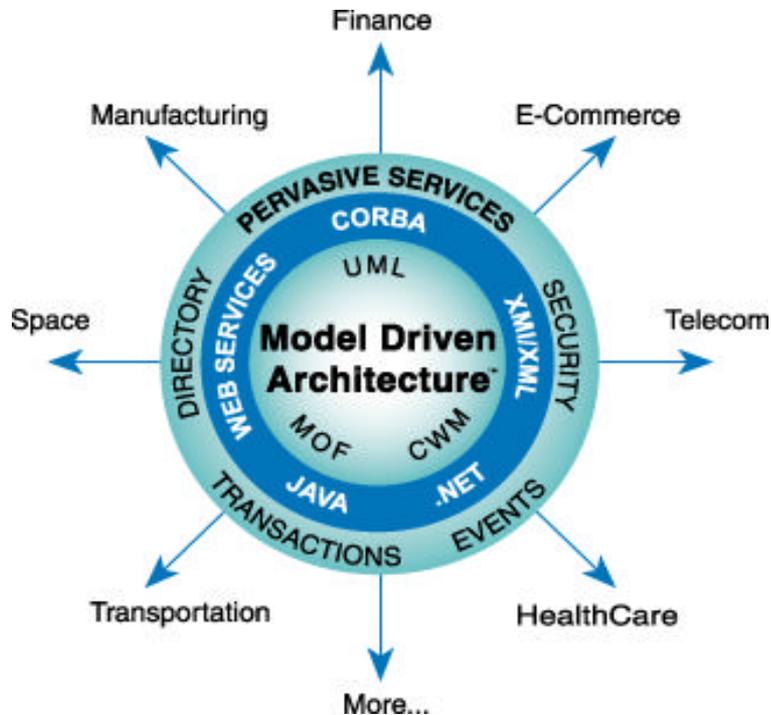
Which MDA Tools are Right for You?



Mike Rosen

CTO, M²VP

Mrosen@m2vp.com

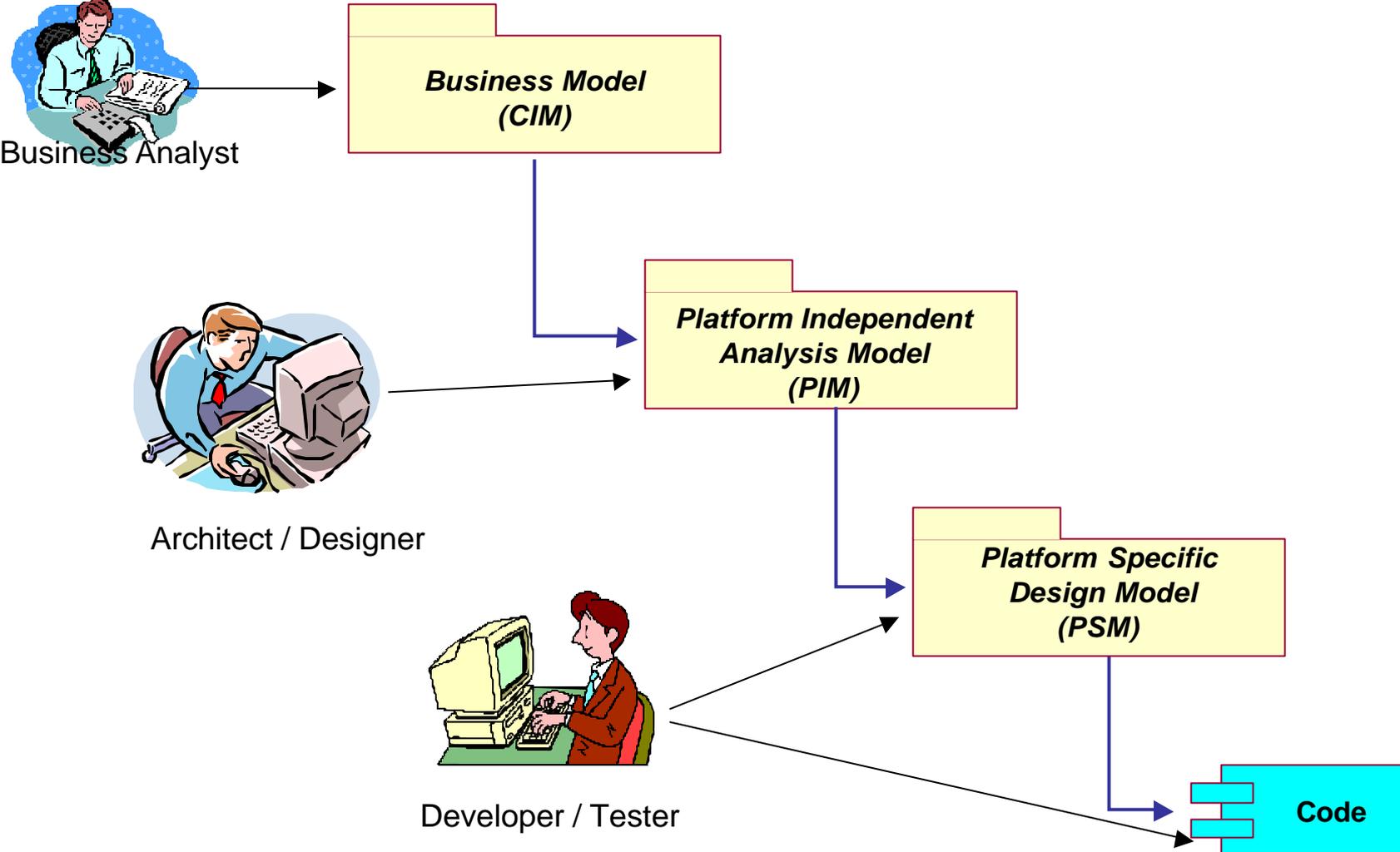


- A set of standards defining the scope, contents, creation and usage of models
- An architecture-based method for integrating models into the development process
- Core Technologies
 - UML + OCL
 - MOF + XMI
 - CWM

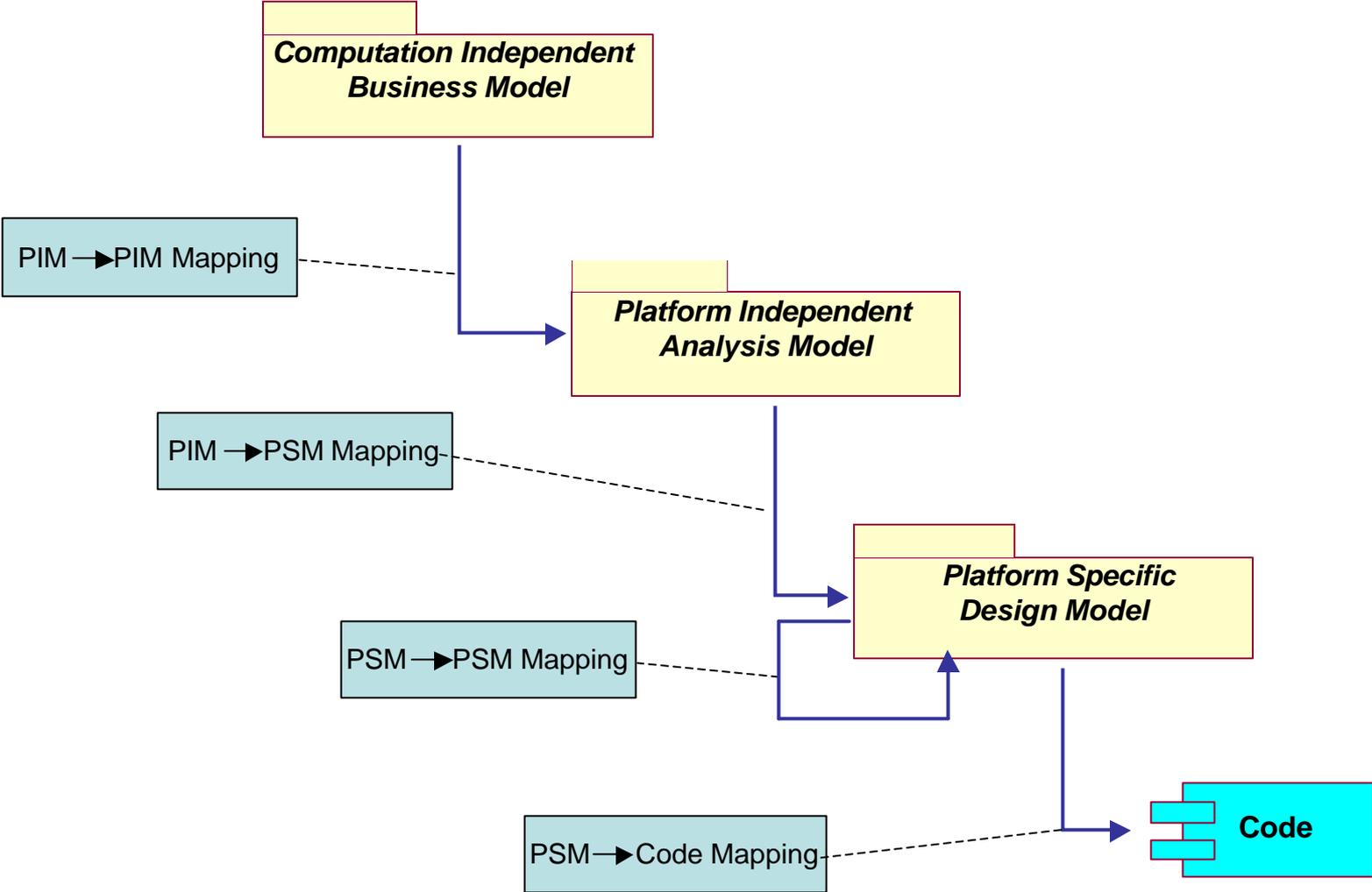
Agenda

- MDA context
- Development process
- Where tools fit in the process
- Requirements for tools
- Integration between tools

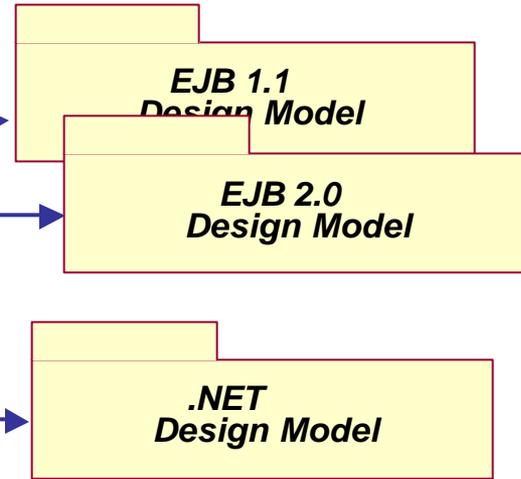
MDA Distilled



MDA Mappings

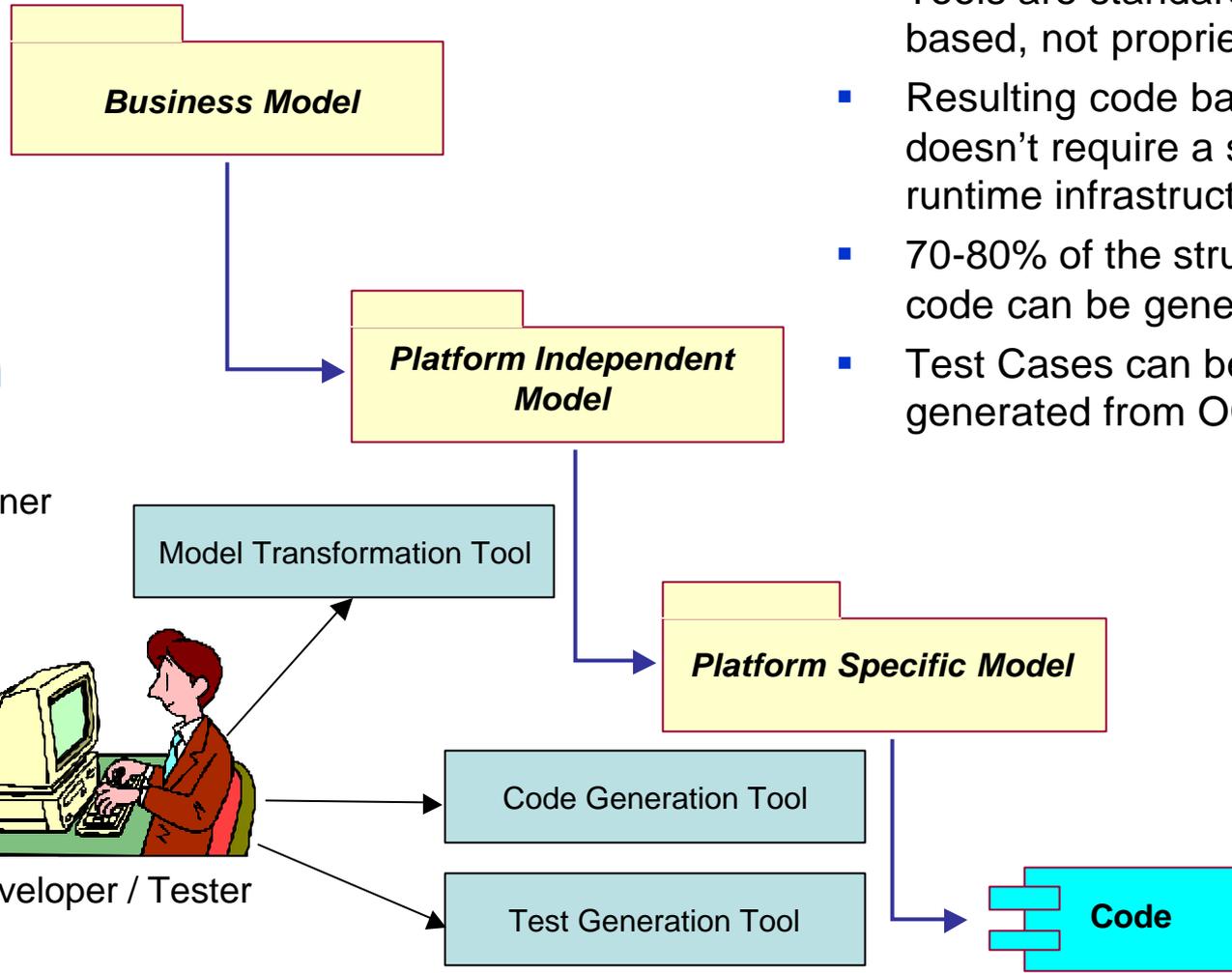
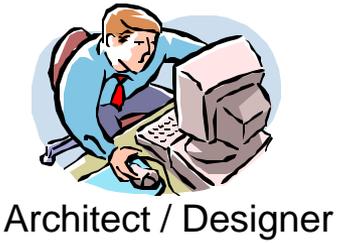


Technology Independence



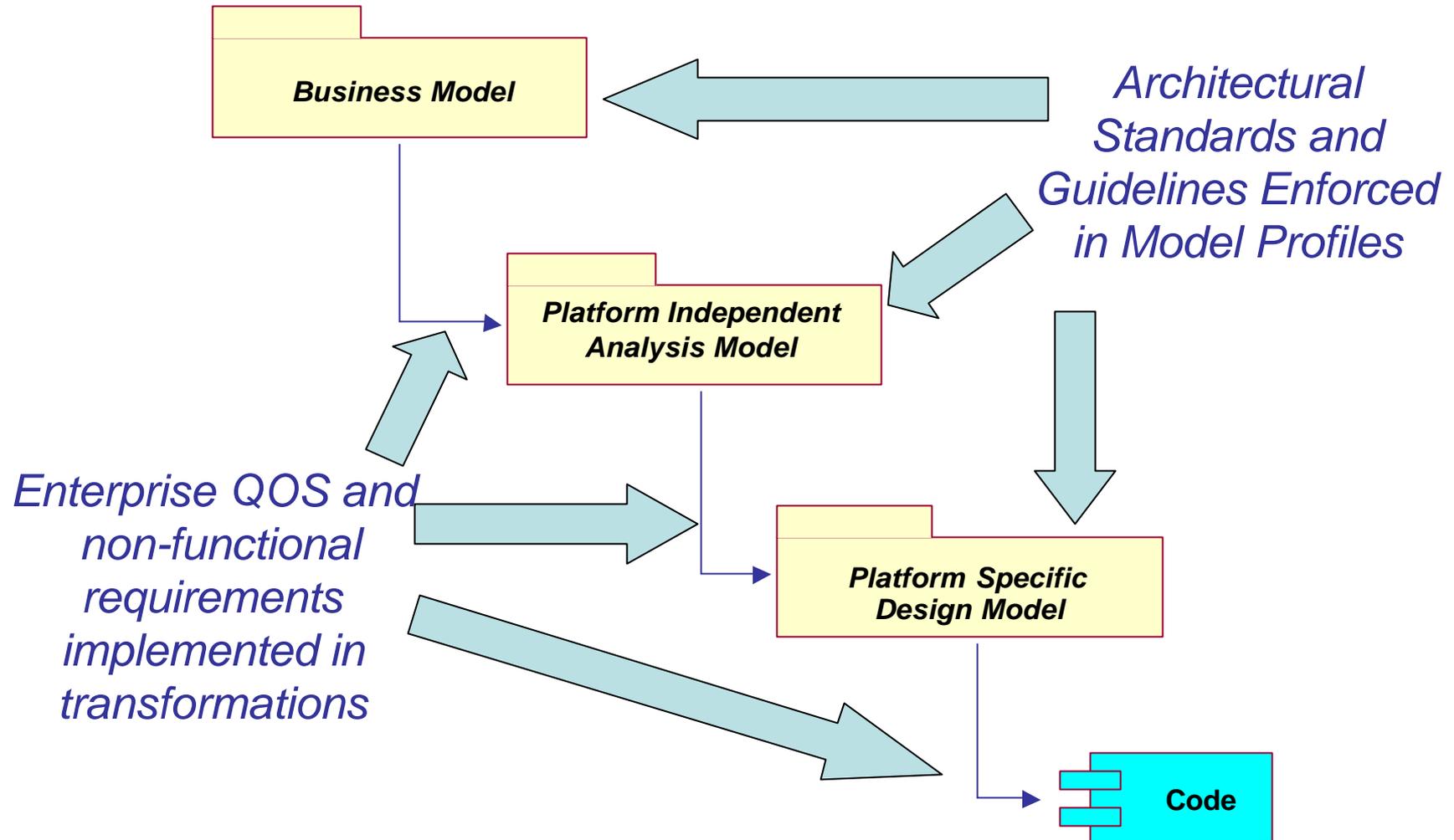
- Applications are “Future-Proof” against technology churn
- When technology evolves, a new PSM can be generated rather than rewriting it
- Artifacts can be generated for multiple platforms from the same design

Generation Tools

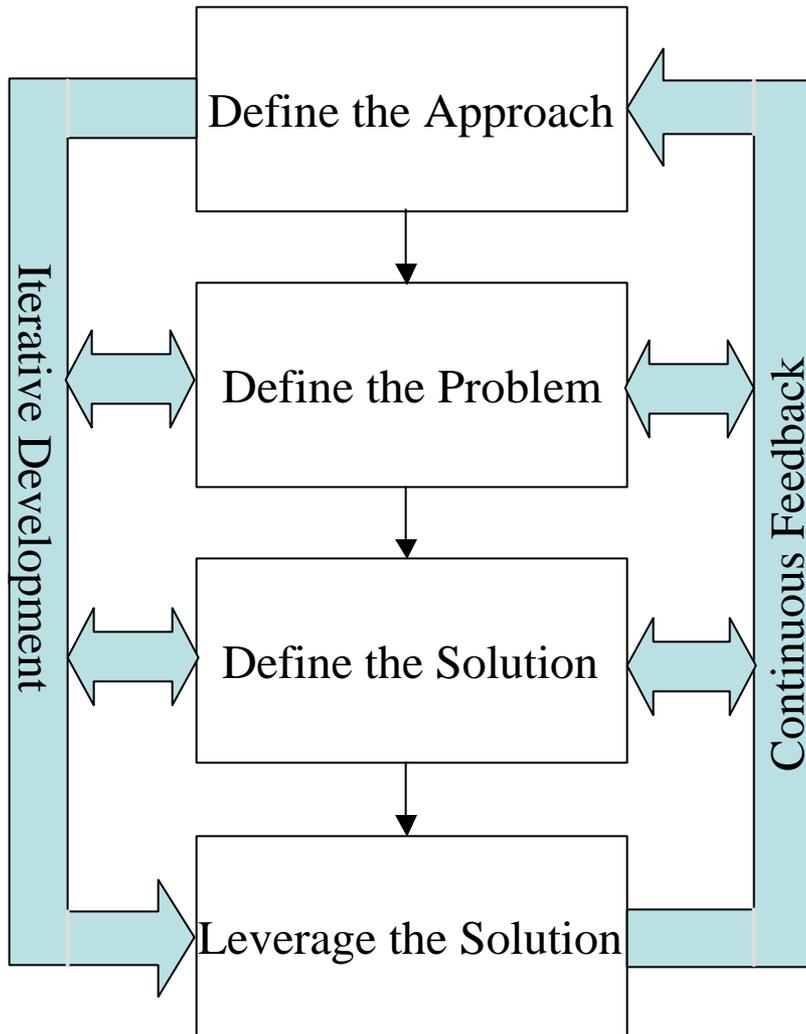


- Tools are standards based, not proprietary
- Resulting code base doesn't require a specific runtime infrastructure
- 70-80% of the structural code can be generated
- Test Cases can be generated from OCL

MDA Under the Hood



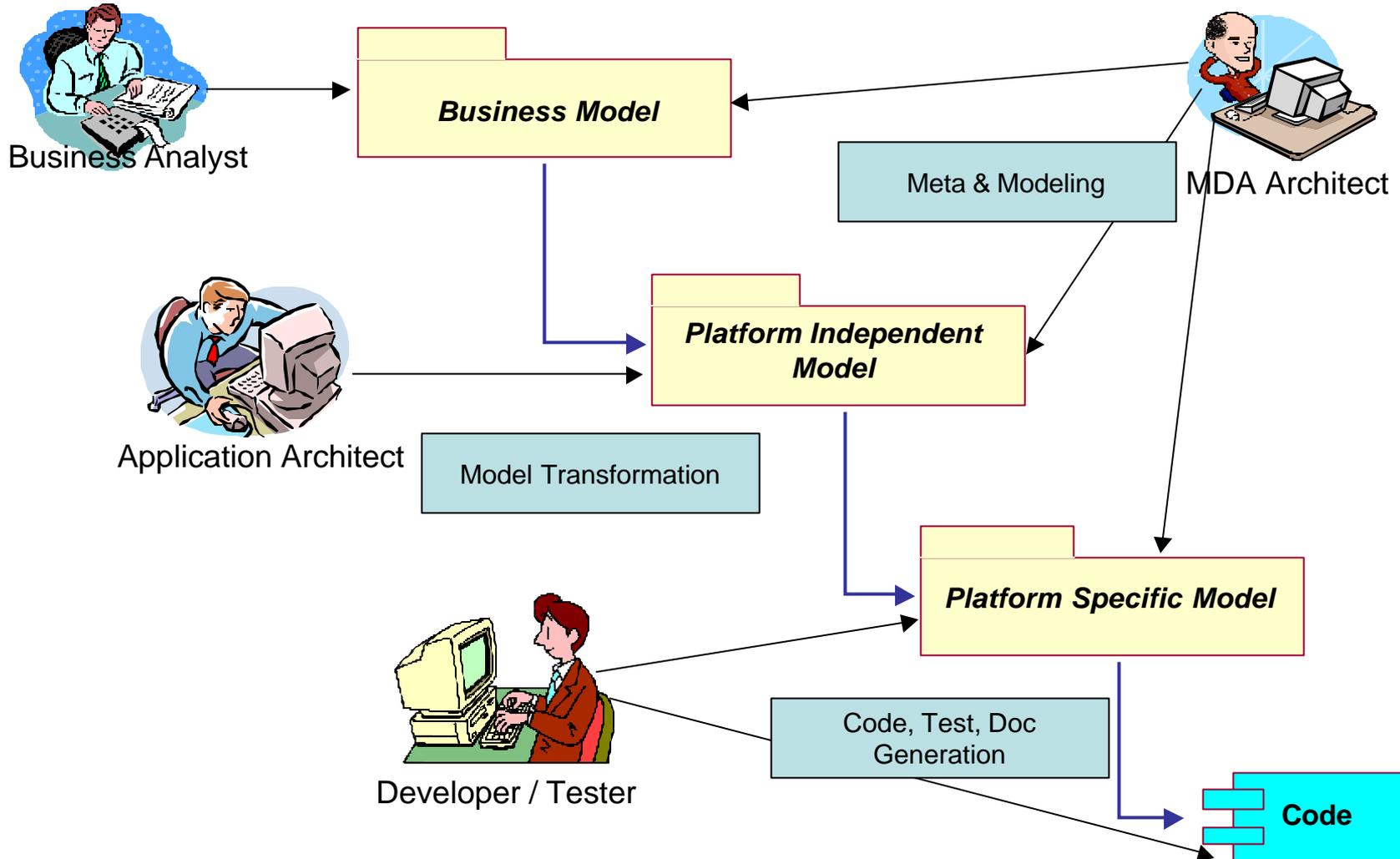
MDA explicitly supports Enterprise Architecture



1. Define the approach
 - Integrate enterprise architecture into the development process.
 - Create meta-models and profiles
2. Define the problem
 - Create Business Models (Domain, CIM, System)
3. Define the solution
 - Refine into PIMs and PSMs
4. Leverage the solution
 - Integrate assets into a reuse repository
 - Architecture and design accommodates: reuse, customization, enhancements, versioning...

- **Computational Independent Model**
 - Simplified UML subset appropriate for business analysts. Non-UML representation.
- **Platform Independent Model**
 - Custom profiles for enterprise architecture and standards
 - Standard based profiles (EDOC, EAI)
- **Platform Specific Model**
 - Standards based profiles (CORBA, EJB, .NET)

MDA Process Review



- Basic desktop tool capabilities
- Ability to create models and meta-models in UML
 - Formally define model constructs and constraints
- Ability to create profiles in UML
 - Since most modeling tools don't support MOF directly, profiles extend standard UML tools
- Ability to import / export meta-models and models in standard XMI format

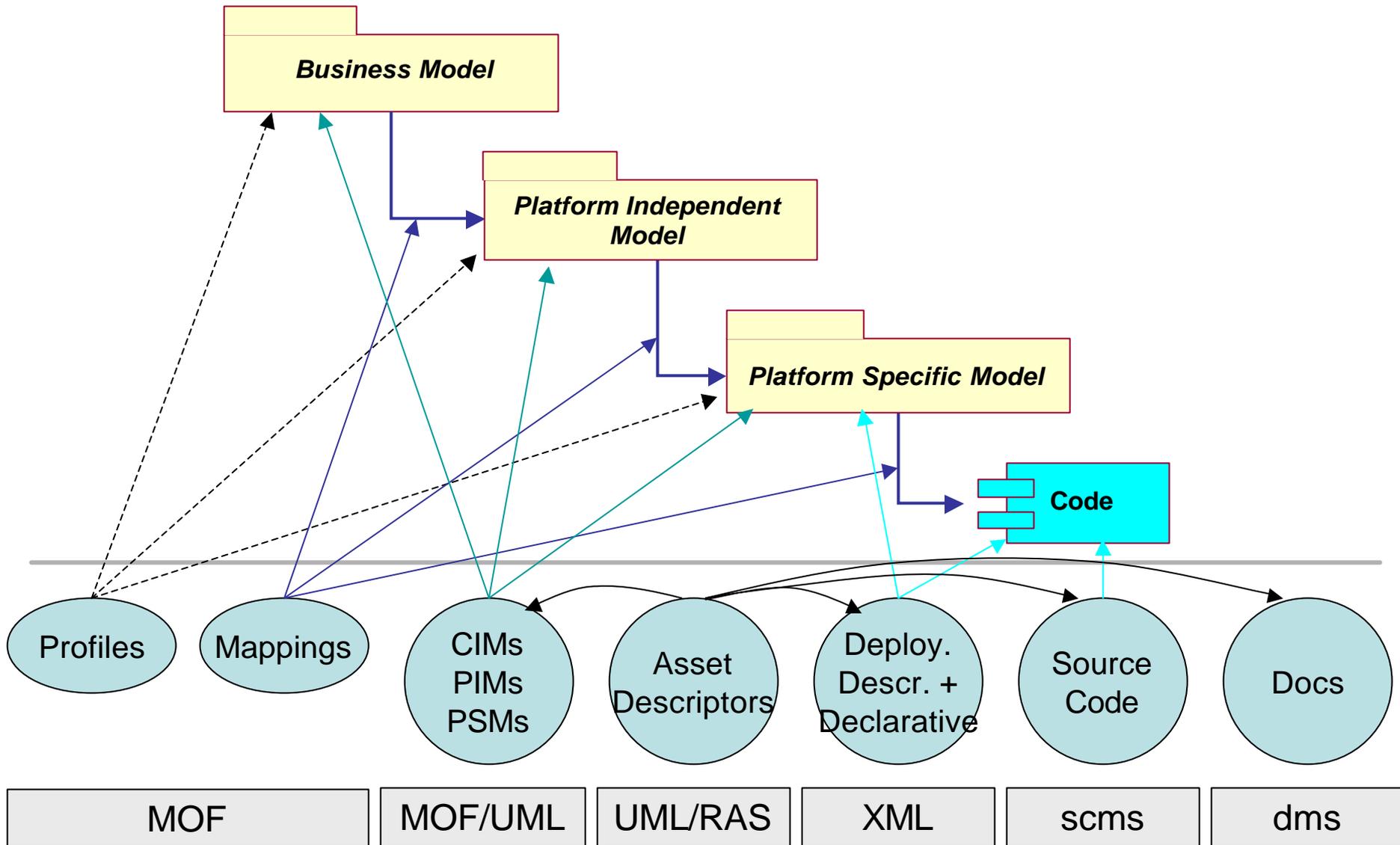
- Ability to formally define mapping between models
- Ability to customize standard mappings
- Ability to parameterize mappings
- Ability to modify transformation result
- Ability to trace elements in one model to the elements they were derived from in another model

- Preserves separation of concerns
 - Platform specific info is not required in PIM
 - Platform independent info is not required in PSM
- Can generate to multiple technologies and platforms
- Generation can be parameterized and modified
- Generation based on standard and custom patterns
- Completeness
 - Structural code (skeletons, etc.)
 - Procedural code
 - OCL can be compiled, generate run-time code
- Can be integrated into an automatic build process (command driven)

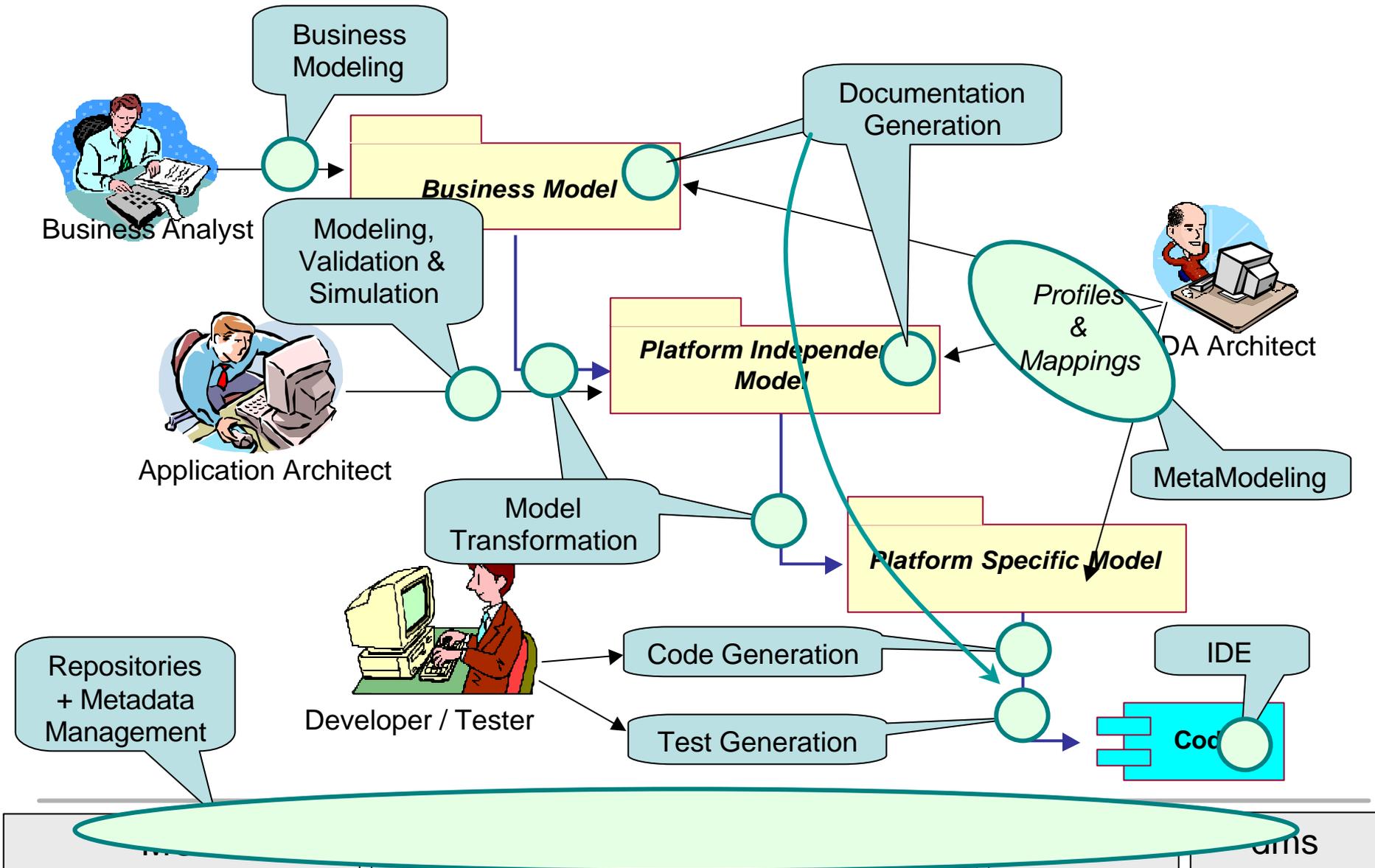
- **Procedural code**
 - Some code can be generated from OCL, but
 - Requires support for Action Semantics
 - Generally necessitates reducing scope of target environment
- **Tests**
 - Can be generated from OCL and Action Language
 - Can be integrated into a framework (e.g. JUNIT, JRUN)
- **Documentation**
 - Ability to create analysis and design documentation from information in all levels of models
 - Ability to customized document and report generation
 - Ability to create some form of user documentation

- Model correctness can be validated against meta-model
- Model can execute in a simulation environment
- Support for versioning of models
- Support for model synchronization
 - Reverse or Round-Trip engineering
- MOF Repository

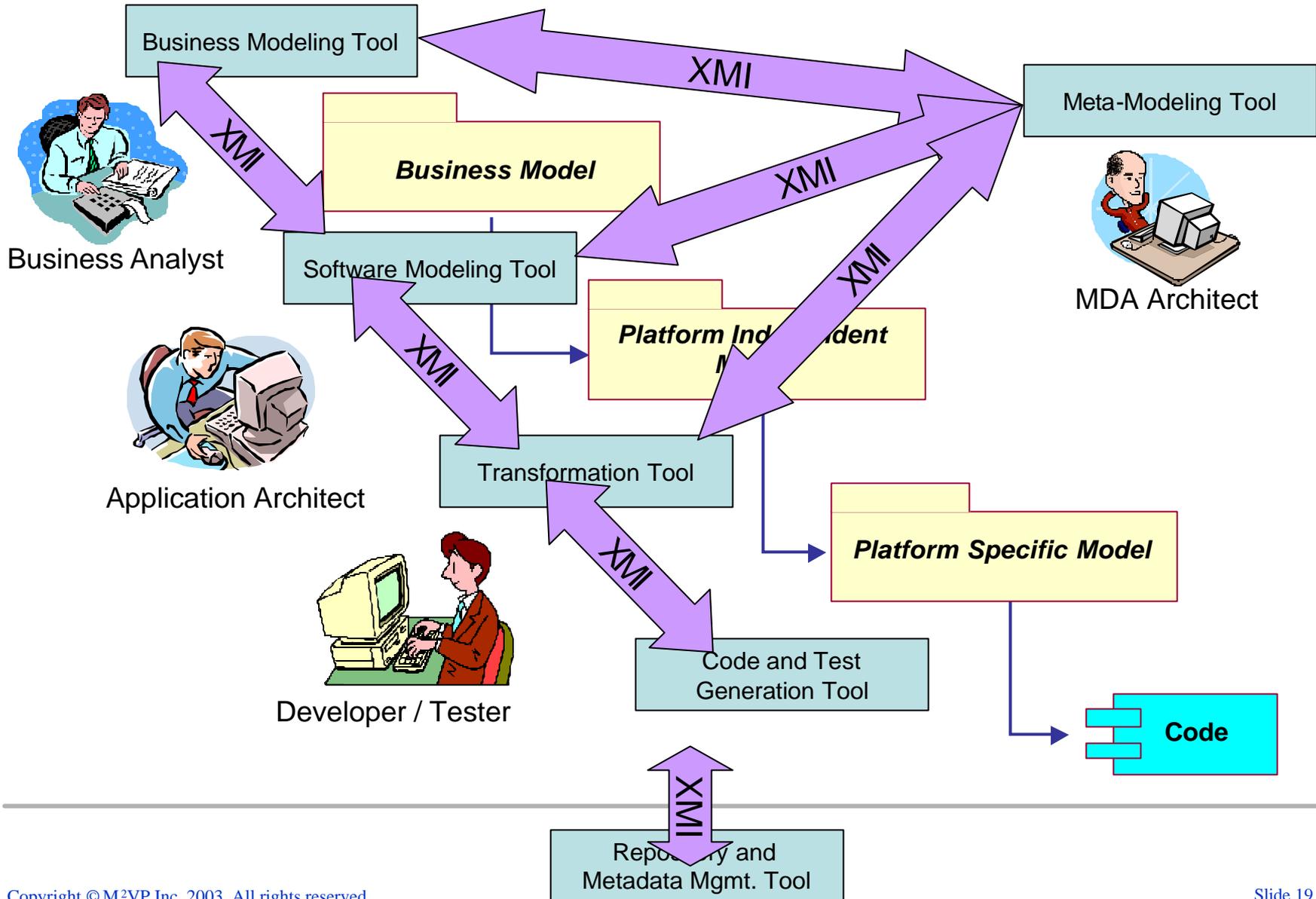
Metadata in the MDA Process



Tools in the MDA Process



MDA Tool Integration



- Meets *YOUR* criteria
- Addresses Full Lifecycle
 - Meta-models, transformations, generation
- Generation
 - SOC, Customization, Completeness
- Advanced Capabilities
 - Synchronization, Versioning, Validation, Simulation, MOF
- Integration with other tools
 - Modeling
 - IDE
 - Repository
- Standards Conformance

- Cost
- Product Architecture
 - Evolvability with the market
 - Support for future platforms
- Documentation, samples, training
- Ease of Use
- Company
 - Support, enhancements
 - Future looks bright
 - Product line is integral
 - Easy to work with

Questions for Your Enterprise M²VP

- What current tools and processes does MDA have to integrate with or support for development, test, reuse, documentation, etc.
- What will the MDA development lifecycle be in the organization?
- Who will perform the business modeling?
- What is the enterprise and application architecture?
- Can these be supported by standard profiles or will the organization be creating custom profiles and metamodels
- What technology platforms will need to be supported?

- For MDA to deliver value
 - It must make models *first class development artifacts*
 - Tools must support all aspects
 - IDE for modeling

- No one tool does it all!
- Enterprises have different requirements
- Integration and customization important
- Standards and tools are critical to MDA success
- Lot's of advancements in the future
 - Action Semantics, MOF 2.0, QVT, RT Engineering
- The market is still young. Expect...
- Continued expansion of MDA tools...
- ... and then contraction of market

Questions

