A THREE VIEWPOINT MODEL FOR SOFTWARE ECOSYSTEMS

John D. McGregor and J. Yates Monteith, School of Computing, Clemson University
Contents

- Brief Introduction
- Background
  - What is a Software Ecosystem?
  - Meta-Modeling
  - Viewpoints and Views
- Ecosystem Modeling
  - Meta-model
  - Derived Three Viewpoint Model
  - Conclusions and Future Work
- Case Study if we have time...?
Who am I?

- [J.] Yates Monteith
  - Manhattan College
    - 2D and 3D Polymer Simulation Research.
  - Clemson University
    - Computer Science Education and Teaching Process.
    - Real Options Value in Software Product Lines.
    - Technical Debt Aggregation in Software Ecosystems.

- Focused on Software Engineering
  - Software Product Lines, Software Ecosystems
  - Software Economics
  - Software Architecture, V&V and more…

- Also, that weird looking guy that spends too much time in 110A…
What is a Software Ecosystem?
What is a Software Ecosystem?

- Good question.
  - What is an ecosystem?
What is a Software Ecosystem?

- Good question.

- What is an ecosystem?

- “A system formed by the interaction of a community of organisms with their environment” [1].
What is a Software Ecosystem?

- Good question.
  - What is an ecosystem?
  - “A system formed by the interaction of a community of organisms with their environment” [1].
- Okay, but how do we relate the idea of ecosystems to software ecosystems?
What is a Software Ecosystem?

- Good question.
- Okay, but how do we relate the idea of ecosystems to software ecosystems?
  - Stems from the idea of a business ecosystem.
What is a Software Ecosystem?

- Good question.
- Okay, but how do we relate the idea of ecosystems to software ecosystems?
  - Stems from the idea of a business ecosystem.
- “Loose network[s] of suppliers, distributors, outsourcing firms, makers of related products or services, technology providers and a host of other organizations” [6].

What is a Software Ecosystem?

- Good question.
- Okay, but how do we relate the idea of ecosystems to software ecosystems?
  - Environment $\rightarrow$ Market Segment
  - Organisms $\rightarrow$ Stakeholders and Players within the Market Segment.
- A Software Ecosystem includes a business ecosystem that centers around a software market!
What is a Software Ecosystem?

- Not quite.
What is a Software Ecosystem?

- Not quite.
- “A software ecosystem consists of the set of software solutions that enable, support and automate the activities and transactions by the actors in the associated social or business ecosystem and the organizations that provide these solutions” [2].

What is a Software Ecosystem?

- “A software ecosystem consists of the set of software solutions that enable, support and automate the activities and transactions by the actors in the associated social or business ecosystem and the organizations that provide these solutions.”

- Software Solutions → A Software Platform
- Activities and Transactions → Development, Extension, Use
- Actors → Users, VARs, 3rd Party Devs
- Organizations → Platform Developers
Examples of Software Ecosystems
Illustrative Example: Windows

- Microsoft Windows exists in the operating system market.
Illustrative Example: Windows

- Some people use Windows because they like Windows, or because Windows is a platform for a software solution they utilize.
Illustrative Example: Windows

- Microsoft (MS) would like to increase their market share. They do this by facilitating application development in third party developers.

- MS User
- Autodesk User
- 3rd Party Dev
- OSX User
- Linux User
Illustrative Example: Windows

- Application Developers target the Window’s platform for application development because of support they receive in development.
Development of attractive applications in Windows increases Microsoft’s market share in the OS market segment.
Illustrative Example: Windows

- This process is facilitated by activities.
  - Release of Libraries and Runtimes
    - .NET, Windows.h, MSVC Runtimes
  - Sponsoring of support methods
    - MSDN, MSDN-AA, Dreamspark, TechNet
  - Implementation of Feature Requests
  - Conferences and Workshops
  - And more...
So what?

- Ecosystems can be on a large scale and very complex.
  - Ecosystems with more than 10K entities and 10K+ relations [11].
  - Complexity of Software + Complexity of Business Relationships.
  - Differing perspectives on the ecosystem.
Eclipse Ecosystem

From March 2008 Eclipse Executive Director's Report:
So what?
Ecosystems can be on a large scale and very complex.

So we need a model and modeling technique that allows us to manage the size of an ecosystem while illuminating the relevant details.

We accomplish that through creation of a meta-model.
Meta-Modeling

- Everyone is familiar with Models.
Modeling

- Everyone is familiar with Models.
  - [Software Engineering] Models are abstractions that communicate information while eliminating unnecessary details. Among other things.
Meta-Modeling

Everyone is familiar with Models.

And we have lots of them.

```
BankAccount

owner : String
balance : Dollars = 0

deposit ( amount : Dollars )
withdrawl ( amount : Dollars )
```
Everyone is familiar with Models.
And we have lots of them.
Everyone is familiar with Models.
And we have lots of them.
Everyone is familiar with Models.
And we have lots of them.
Everyone is familiar with Models.

And we have lots of them.
Meta-Modeling

- But what do we use to define a model?
Meta-Modeling

- But what do we use to define a model?
- Answer: A meta-model.
Meta-Modeling

- Models describe a set of entities and the relationships between them.

- Meta-Models describe the types of entities that can exist within a model and the relationships that can exist between them.
Within software engineering, numerous meta-models exist.

- Object Management Group’s Software Process Engineering Meta-model
- ISO/IEC 24744 Meta-model for Development Methodologies
So models are abstractions that help us visualize the most essential parts of a problem context.

But not every model is suitable for every stakeholder.
So models are abstractions that help us visualize the most essential parts of a problem context.

But not every model is suitable for every stakeholder.

And different stakeholders might have different perspectives on the same model...
Models, Views and Viewpoints...
So, given multiple stakeholder perspectives, we define multiple viewpoints for a given model.

Viewpoints are designed to target groups of stakeholders with differing concerns.

And from a defined viewpoint, a view is instantiated that represents the concrete notion of that viewpoint.

We can also think of it programmatically:

- Language $\Rightarrow$ Type $\Rightarrow$ Instantiation of type
- Model $\Rightarrow$ Viewpoint $\Rightarrow$ View

The Meta-Model
Three Viewpoints

- Business Viewpoint
- Software Viewpoint
- Innovation Viewpoint
Business Viewpoint

- The business viewpoint encapsulates the organizations and their relationships that are involved within the software ecosystem into a single view.
  - Suppliers, Competitors, Complimenters, Customers, Users.
  - Standards Organizations, Governance Boards.
Porter’s Five Forces for Strategy Development

- Five competitive forces that guide business.
- Assists in identification of organizations that engage in Porter’s roles.

Porter’s Five Forces for Strategy Development

- Five competitive forces that guide business.
- Assists in identification of organizations that engage in Porter’s roles.

- Strategic Managers
- Technical Managers
- Corporate Officers

Business Viewpoint

- Transaction/Transfer Analysis

- Used to develop models of the supply chains and value chains attributed business relationships.
  - **Transactions** represent a cost-associated exchange of assets between two organizations.
  - **Transfers** represent a cost-free exchange of assets between two organizations.

The Software Viewpoint models the slice of the ecosystem that is concerned with technical assets.

Modeled by a “shared uses” and “exclusive uses” relationship between software assets.

The relationships form a supply chain of components and libraries that are utilized by a particular software asset.

Targeted Stakeholders:
- Developers
- Testers
- Architects
- Test Managers
- Project Managers
Innovation Viewpoint

- Four types of innovations encompassed by this view [10]:
  - Product Innovations
  - Process Innovations
  - Experience Innovations
  - Business Model Innovations

- Modeled by entities which are the result of innovation and relationships between software assets and organizations that drive innovation.

- Targeted Stakeholders:
  - Strategic Managers, Corporate Officers

So what’s the point?

- Software ecosystems can get very big.
  - Known from recent experience with ecosystems with more than 10,000 organizations.
So what’s the point?

- Software ecosystems can get very big.
  - Known from recent experience with ecosystems with more than 10,000 organizations.
  - By creating a meta-model, we have a more formalized basis for the creation of models to represent ecosystems.
So what’s the point?

- Software ecosystems can get very big.
  - Known from recent experience with ecosystems with more than 10,000 organizations.
  - By creating a meta-model, we have a more formalized basis for the creation of models to represent ecosystems.
  - By defining viewpoints and instantiating views, we are able to create some more manageable models.
So what’s the point?

- Software ecosystems can get very big.
  - Known from recent experience with ecosystems with more than 10,000 organizations.
  - By creating a meta-model, we have a more formalized basis for the creation of models to represent ecosystems.
  - By defining viewpoints and instantiating views, we are able to create some more manageable models.
  - Manageable models allow us to isolate the interesting parts of the ecosystems.
So what’s the point?

- Software ecosystems can get very big.
  - Known from recent experience with ecosystems with more than 10,000 organizations.
  - By creating a meta-model, we have a more formalized basis for the creation of models to represent ecosystems.
  - By defining viewpoints and instantiating views, we are able to create some more manageable models.
  - Manageable models allow us to isolate the interesting parts of the ecosystems
    - Notably the interactions with other organizations and the software supply chains.
Software View: Eclipse Ecosystem

- Source ➔ Package dependency graph
- 2.4k nodes, ~20k edges
Eclipse Ecosystem View: W3C / XML cluster

- 70 nodes, ~100 edges.
Eclipse Platform View:

IBM cluster

- 200 Nodes, ~200 edges
Eclipse Platform View: Ant cluster

- ~70 Nodes, 100 edges
Eclipse Platform View: Akrogen cluster

Linux User
Conclusions & Future Work

- Created a meta-model for software ecosystems that allows for multiple definition of multiple viewpoints.
- Defined three viewpoints that encompass software ecosystems holistically, based on both experience and literature.
- What’s next?
Conclusions & Future Work

- Created a meta-model for software ecosystems that allows for multiple definition of multiple viewpoints.
- Defined three viewpoints that encompass software ecosystems holistically, based on both experience and literature.
- What’s next?
  - Model ecosystems
  - Perform analyses
  - Generalize analyses
References


Case Study: Leducq Trans-Atlantic Single Ventricle Modeling (SVM) Project

- Five year multi-disciplinary globally-distributed medical research grant.
- Focuses on collecting data, theoretically modeling and experimentally verifying a set of procedures concerned with correcting Hypoplastic Left Heart Syndrome.
- Incorporates multiple organizations, multiple domains and multiple software products.

Nota Bene: I worked on this project. See http://modelingventricle.clemson.edu
Leducq SVM Project

- Three medical institutions that collect data and upload it to a repository:
  - MUSC, Great Orman Street Children’s Hospital, University of Michigan.
Leducq SVM Project

- Three medical institutions that collect data and upload it to a repository:
  - UCSD, Politecnico Milano, INRIA
Three medical institutions that collect data and upload it to a repository:

Three bio-modeling institutions that create theoretical models from data:

Clemson University mechanical engineers create apparati that experimentally verify the theoretical models.
Leducq SVM Project

- Three medical institutions that collect data and upload it to a repository:
- Three bio-modeling institutions that create theoretical models from data:
- Clemson University mechanical engineers create apparatus that experimentally verify the theoretical models.
- Clemson University computer scientists that manage data repository, webpage, database and perform application development.
SVM Analyses: Business View

- **Suppliers:**
  - SimTK.org, Parasolid (Libraries)
  - Drupal

- **Buyers**
  - Research communities.
  - Commercialization possibilities.

- **Substitutes**
  - Substitutes exist, but not as full featured as SimVascular.
  - Open-Source orgs are working to produce FOSS versions of commercial libraries.

- **Competitors**
  - Other research groups competing for grant dollars.

- **Potential Entrants**
  - None identifiable at this time.
SVM Analyses: Business View

- Leducq Foundation
- MUSC
- UMich
- INRIA
- SVM Cluster
- Clemson University
- Politecnico Milano
- UCSD
- Simtk.org
- GOS Hospital
- OpenSourceOrg
SVM Analyses: Software View

- **Two Clusters**
  - Webpage Portal and Data Repository
    - Utilizes Drupal with MySQL
    - Matlab and for model parameter generation.
  - SimVascular Cluster
    - SimTK.org SimVascular product
    - Three Commercial Libraries: Parasolid, Meshsim, openVascular.
    - One OSS Library: openVascular
    - Utilizes Intel C++/Fortran95 Compilers.
SVM Analyses: Software View

- Parasolid
- Meshsim
- SimVascular
- Intel Fortran
- Intel C/C++
- LesLib
- openVascular

- MySQL
- Drupal
- MatLab model
- Mimics

Diagram shows relationships between software components, indicated by "uses" and "sharedUses" connections.
Analyses

- Business View
  - Some organizations are not nearly as connected as others.
    - UCSD and INRIA form the bridge between much of the software and the rest of the ecosystem organizations.

- Software View
  - Two single points of failure exist.
    - 1) Drupal for the Web Portal and Data Repository
    - 2) Commercial Libraries for SimVascular
Extra Stuff: Tomcat