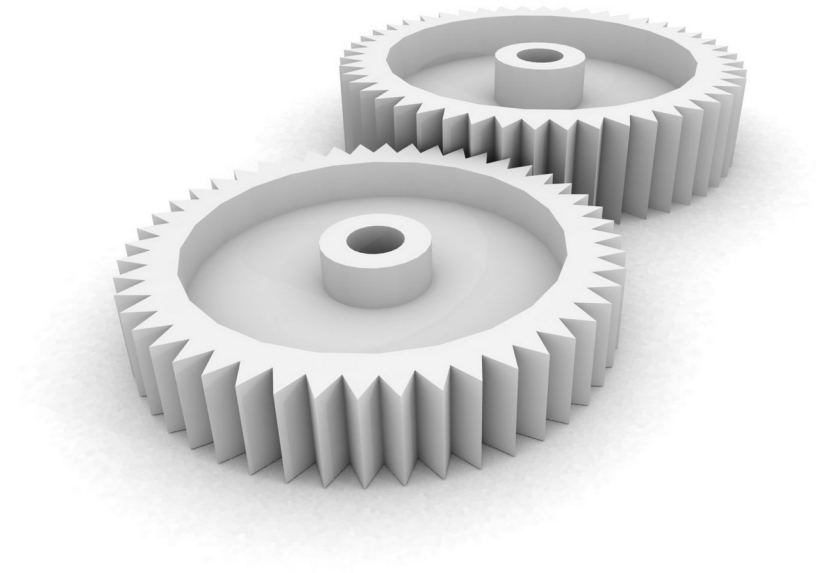


Mylyn meets **Intent** :

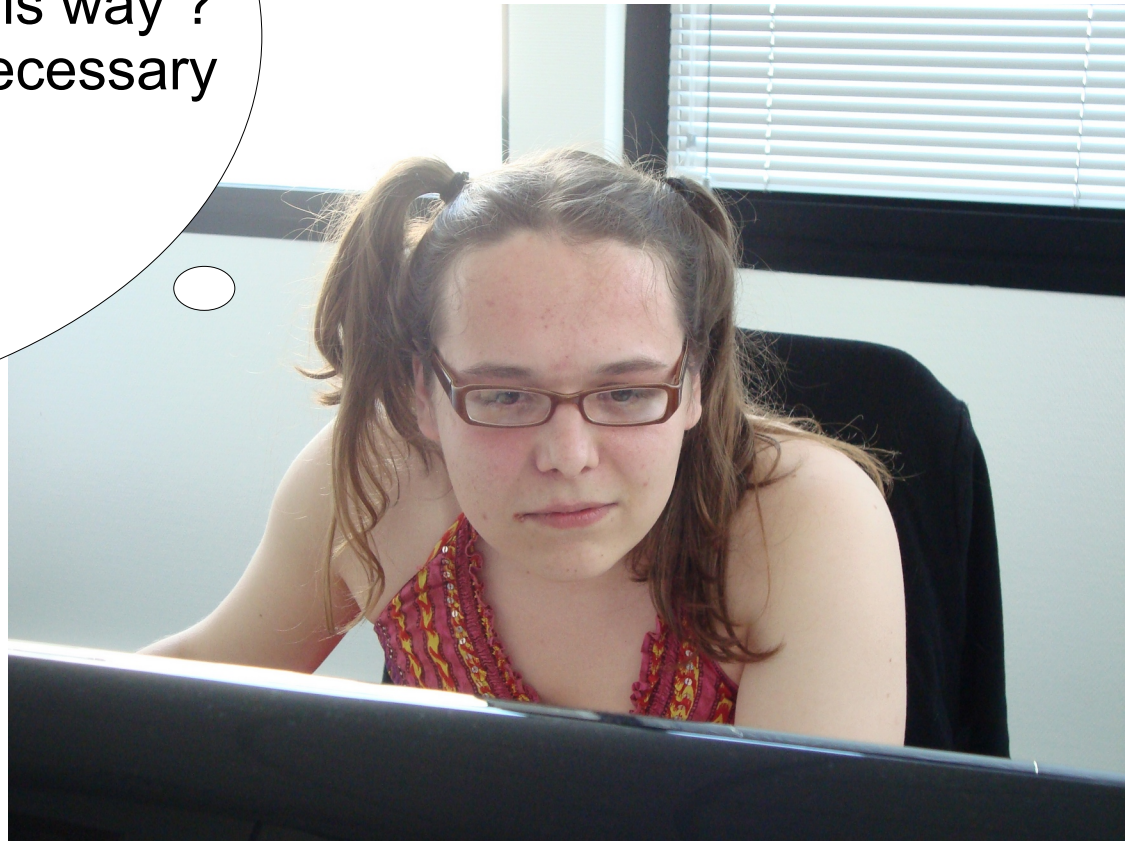
Documentation made **fun** and **useful**

Cédric Brun  
Obeo, France



# Let's meet **Alice**

Hey, why is this thing designed this way ?  
It looks way more twisted than necessary



# Bob's answer

Just check the 3 years old design document !



# But we already have **Doc !**

- EMF Compare Guide
  - User Guide
- Developer Guide
  - Architecture
  - Using The Compare Services
  - Customizing Compare Services
  - Adding new actions to the export menu

- diff
  - DiffModel
    - GenModel
    - getSubchanges() : Elnt
    - getDifferences(EObject) : DiffElement
    - getDifferences() : DiffElement
    - ownedElements : DiffElement
    - leftRoots : EObject
    - rightRoots : EObject
    - ancestorRoots : EObject
  - DiffResourceSet
    - GenModel
    - getSubchanges() : Elnt
    - diffModels : DiffModel
    - resourceDiffs : ResourceDiff
  - DiffElement
    - GenModel

This packages backs up EMF Compare's matching functionality.

See:

[Description](#)

## Interface Summary

<a href="#">MatchOptions</a>	Defines constants for the different options available to tweak the matching process.
------------------------------	--

## Class Summary

<a href="#">EMFCompareMatchMessages</a>	Utility class to access externalized Strings for EMF Compare's match.
<a href="#">MatchPlugin</a>	The activator class controls the plug-in life cycle.

## Package org.eclipse.emf.compare.match Description

This packages backs up EMF Compare's matching functionality.

It provides services to call for the matching process on two or three EObjects, allowing for

- Content matching on random EObjects
- Model matching with model roots
- Resource matching with model resources
- ResourceSet matching on whole resource sets

All the features provided by the match service can be called stand-alone.

# **We Want a doc**

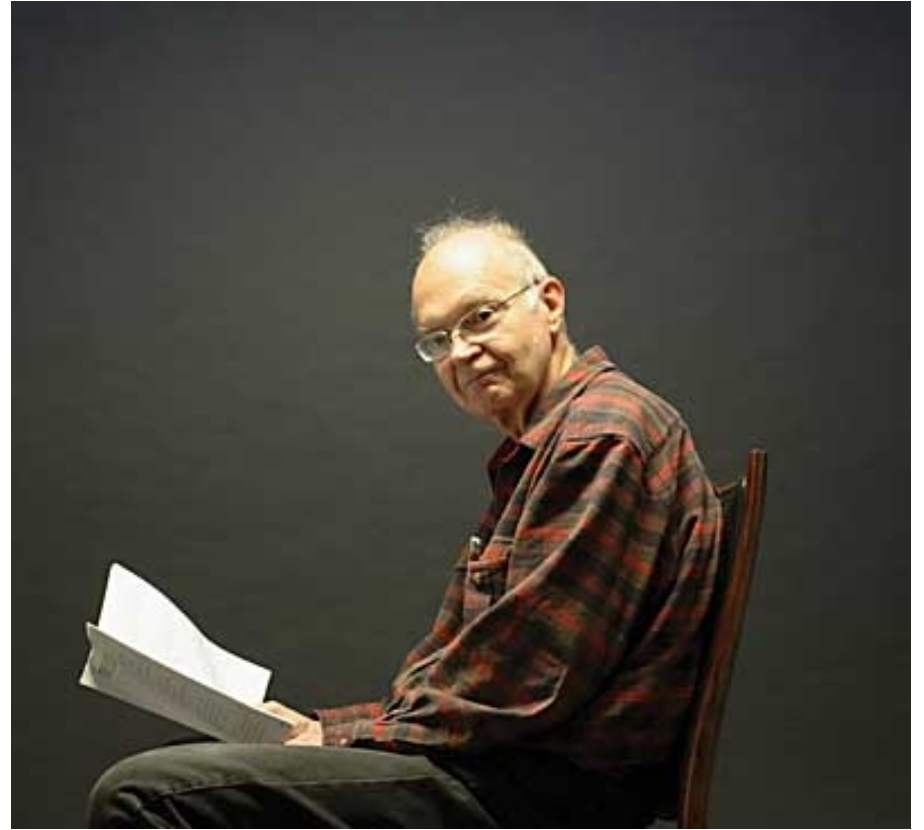
easy to access

easy to maintain

esthetically appealing



# Powered by **Literate Modeling**



# Mixing documentation and Model Fragments

**Section {**

## Architecture of the match package

The Match metamodel can be found at the following location :

```
@M matchResourceDefinition
    Resource matchResource {
        URI = "platform:/resource/org.eclipse.emf.compare.match/model/match.ecore";
        contentType = "ECore";
        content += match;
    }
```

**Mc**

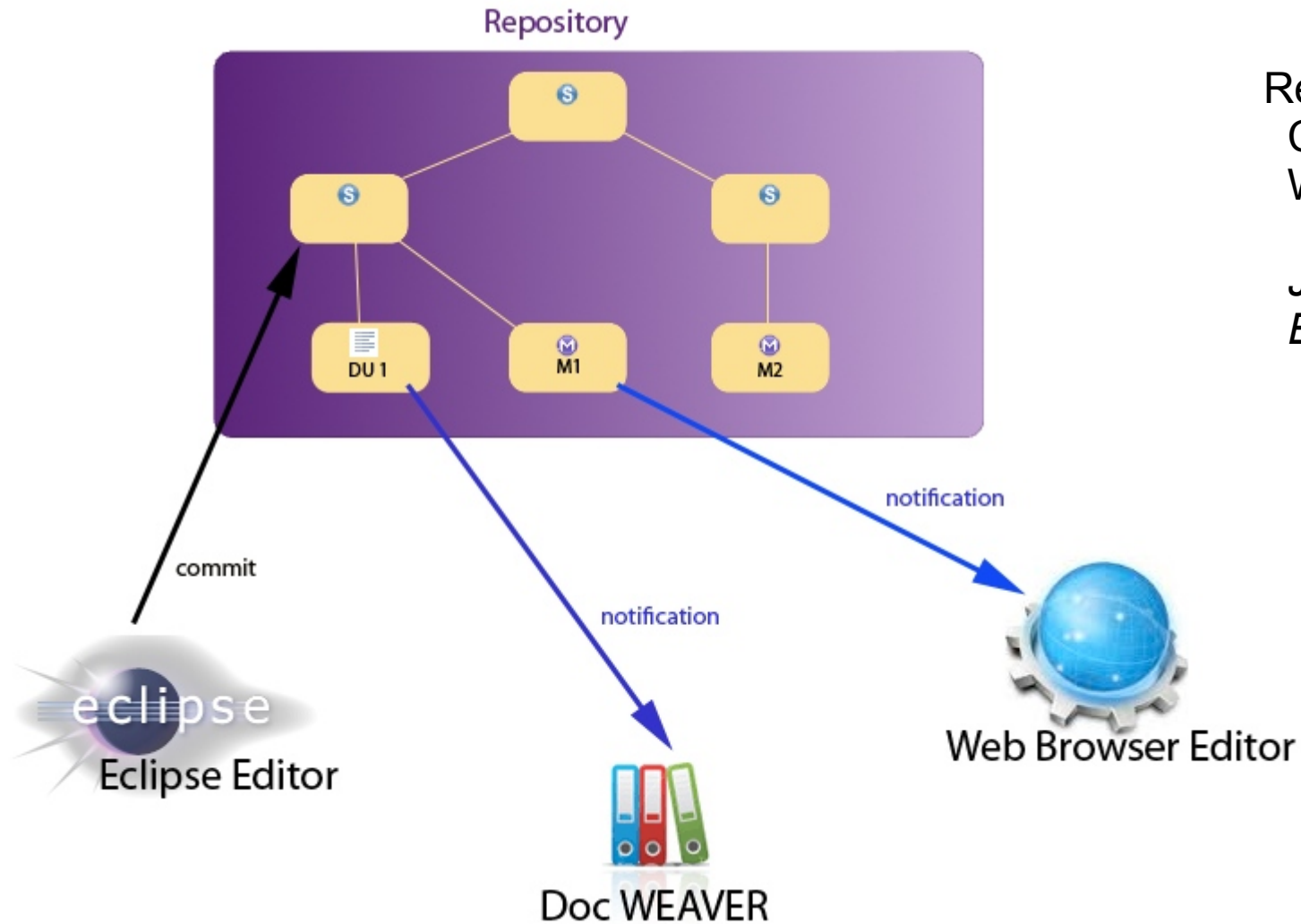
All concepts are defined in the **\*match\*** EPackage, which is associated to a standard eclipse URI :

```
@M matchPackage_definition
    new EPackage match {
        nsURI = "http://www.eclipse.org/emf/compare/match/1.1";
        nsPrefix = "match";
    }
```

**Mc**

}

# Collaboration wired in



Repository might be :  
CDO  
Workspace

*JGit ?*  
*ECR ?*



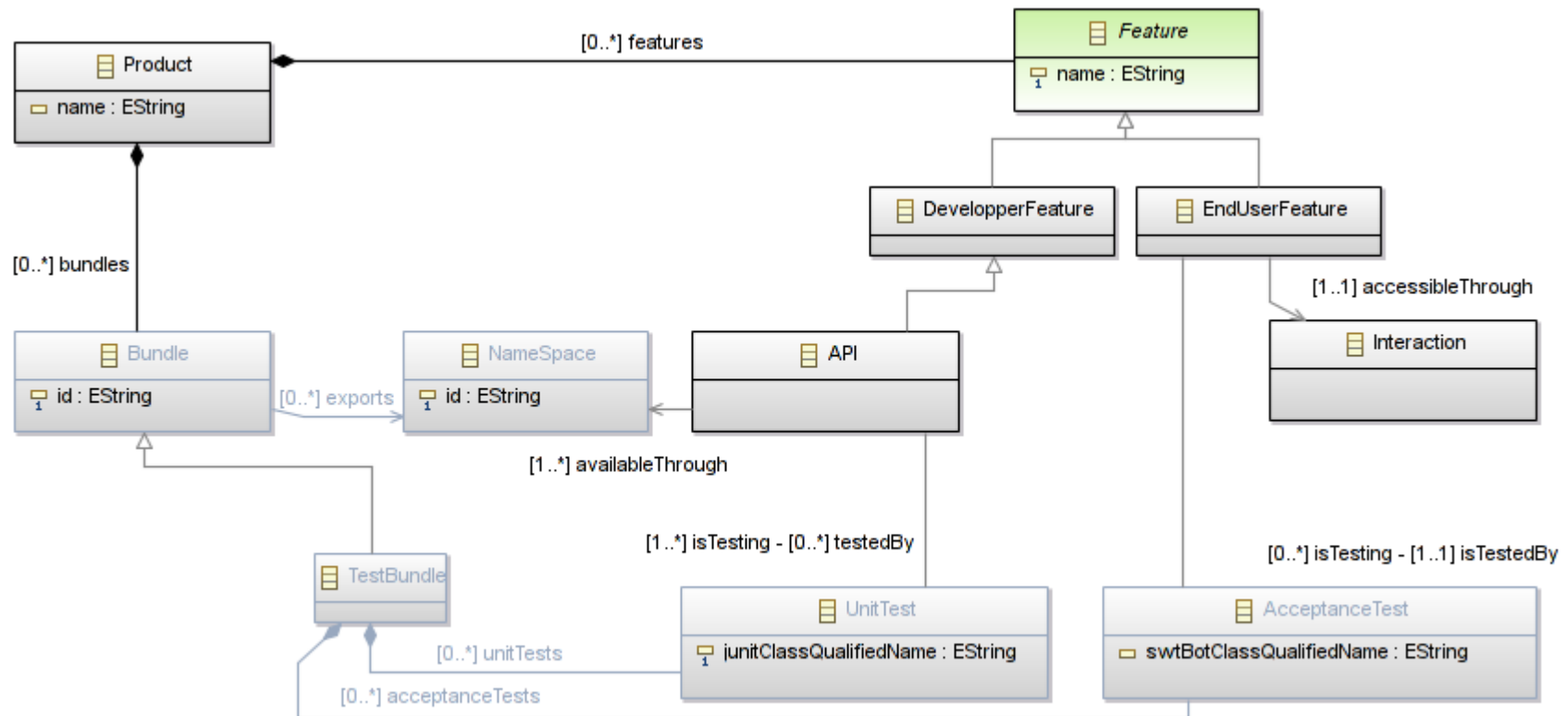
# everything is a **Client**

the editor is a client displaying the document in textual form

the compiler is a client listening to model fragment changes

# Extending Intent

*dedicated domain model*



# Extending Intent

*Validation rule*

*Synchronization Logic*

# Extending Intent *enablement*

The `*org.eclipse.emf.compare.tests plugin*` contains all tests relative to EMFCompare features.

```
@M
Resource compareTestProjectResource {
    URI = "retro:/org.eclipse.emf.compare.tests";
    content += emfCompareTestProject;
}
new Project emfCompareTestProject {
    id = "org.eclipse.emf.compare.tests";
}
MC
```

URI Schemes and protocol to query the workspace

# What's **Next** ?

March : project being accepted and provisionned

June : early adopters build

Q3 : 0.7.0 release

Functional Workspace Backend

Eclipse IDE textual client

Model Synchronization

# Want More ?

[Page](#)[Discussion](#)[View source](#)[History](#)[Edit](#)

## Intent

### Presentation

[edit](#)

Intent is a proposed new open source project under the [Mylyn](#) top-level project. The project is currently in the [Proposal](#) phase. The [proposal](#) has been published and is waiting for comments.

The main purpose of Intent is to allow developers to easily write a documentation up-to-date with any development artifact (models, code, environment...). The documentation, integrated with other Eclipse tools, then becomes truly useful and is no longer a burden that developer have to carry to please their customers or their project manager.

Based on Donald Knuth's Literate Programming concepts and allowing collaborative work around design tasks, Intent will be presented at [Eclipse Con 2011](#).

*"Let us change our traditional attitude to the construction of programs: Instead of imagining that our main task is to instruct a computer what to do, let us concentrate rather on explaining to human beings what we want a computer to do."*

Donald Knuth, Literate Programming (1984)

### Intent Overview

[edit](#)

The following pages provide an overview of the Intent project :

- [Project mission statement](#) - general presentation of Intent
- [Use Cases and Scenarios](#) - step by step use cases with screenshots
- [Architecture](#) - design choices and extensibility
- [June 2011 release plan](#) - what we intend to do for the first Intent release
- [Buzz](#) - articles and blog posts about Intent

### The Intent Project

[edit](#)

Intent is an Eclipse based product created and developed by the Eclipse Strategic Member [Obeo](#). It is meant to be aligned with the Eclipse release train of Eclipse 3.7 (Indigo).

It is meant to be released under the [Mylyn](#)'s Docs project.