

Introduction to RepOSS

October 18, 2011
Northeast Asia OSS Promotion Forum WG3

Background and Goal

Surrounding OSS

1. In many ICT fields, OSS (Open Source Software) is having important roles as key components of its system
2. As of July, 2011, about 260K OSS projects with 2.7M developers were registered into sourceforge.net (From sourceforge.net)
3. In order for stakeholders to choose OSS as to which one is more suitable for their systems, OSS information under some criteria will be useful as a reference

Concerns around OSS

- Functions, Supportability, and Quality
- Continuity, Stability of its community
- License restrictions and IPR
- POC (Proof of Concept), Show cases

Our Goal

To promote OSS adoptions and applications by providing a practical OSS reference and assessments with respect to concerns above

Existing OSS Assessment Methodologies

OSS Assessment Methodologies

	OSMM Capgemini	OSMM Navica	QSOS	OpenBRR	OMM
Seniority	2003	2004	2004	2005	2008
Original authors/sponsors	Capgemini	Navicasoft	Atos Origin	Carnegie Mellon Silicon Valley, Spike Source, O'Reilly, Intel	Qualipso project, EU commission
License	Non-free license, but authorized distribution	Assessment models licensed under the Academic Free License	Methodology and assessments results licensed under the GNU Free Documentation License	Assessments results licensed under a Creative Commons license	Assessments results licensed under a Creative Commons license
Assessment Mode	Practical	Practical	Practical	Scientific	Scientific
Detail levels	2 axes on 2 levels	3 levels	3 levels or more	2 levels	3 levels
Predefined criteria	Yes	Yes	Yes	Yes	Yes
Technical/functional criteria	No	No	Yes	Yes	Yes
Score model	Flexible	Flexible	Flexible	Strict	Flexible
Scoring scale by criterion	1 to 5	1 to 10	0 to 2	1 to 5	1 to 4
Iterative process	No	No	Yes	Yes	Yes
Criteria weighting	Yes	Yes	Yes	Yes	Yes
Comparison	Yes	No	Yes	No	No

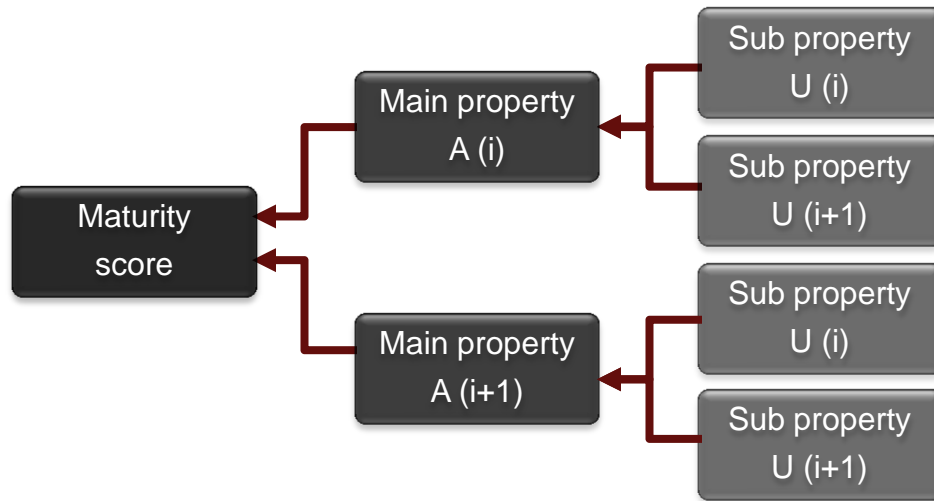
※from Wikipedia, July, 2011

Yet no standard, as the case may be

OpenBRR	No open repository
	Terminology is broad and imprecise
QSOS	Ambiguous scoring rules
	Universality of scoring rule is not possible
Qualipso OMM	Criteria based on Waterfall model
	No open repository

- 1. Deprez, J.-C., Alexandre, S., "Comparing Assessment Methodologies for Free/Open Source Software: OpenBRR & QSOS"**
- 2. Etiel Petrinja, Alberto Sillitti, and Giancarlo Succi, "Comparing Open BRR, QSOS, and OMM Assessment Models"**

Common ground of Evaluation Methods



$$M = \sum_{i=1}^n A_i \times W_i$$

$$A = \left(\sum_{i=1}^n U_i \times W_i \right) \div n$$

Maturity score:

A_i : the score of main property i ;

W_i : the weight of main property i .

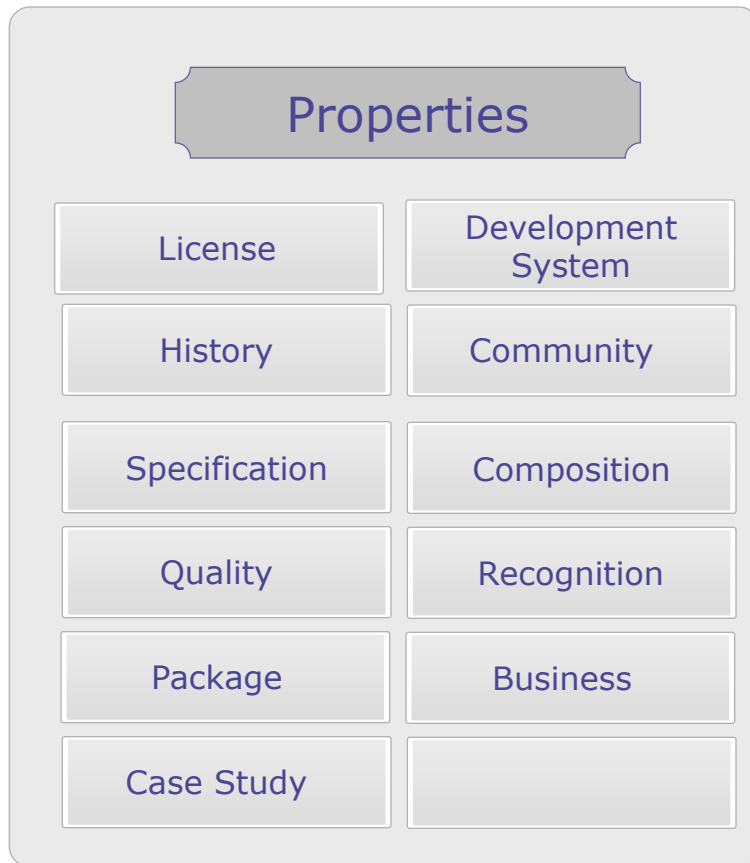
U_i : value of sub property i .

W_i : the weight of sub property i in the property class. 'n' is number of properties in the property class.

Difficulties to overcome

1. Subjective Weight assignment
2. Determination of critical maturity score
3. Different results from various OSS maturity assessment methodologies
4. Objection from OSS community (i.e. Meanings of its summation)

Issues with Property Collection



- 1. Deviation** by Subjective information and/or qualitative analyses
2. Deep and detailed analyses with communities members without **objections from OSS communities**
- 3. Continuity** and the latest information

Existing Methodologies are:

- Using similar properties and similar equations, but
- Have weights that would be depended on assessors' perspectives
- Qualitative analyses and Subjective evaluation are creating deviations



No Universal Standard

Concept and approach of



**Which is a flexible OSS
Assessment Repository**

Approach toward our Goal

- No universal Standard
- Subjective or Qualitative information
- Continuity issue

1. OSS Repository for Flexible assessments

2. Objective and Quantitative Properties

3. Open Community Model

Properties (Quantitative information)



Assessor's Perspectives and Purposes



Methodologies (QSOS, OpenBRR)

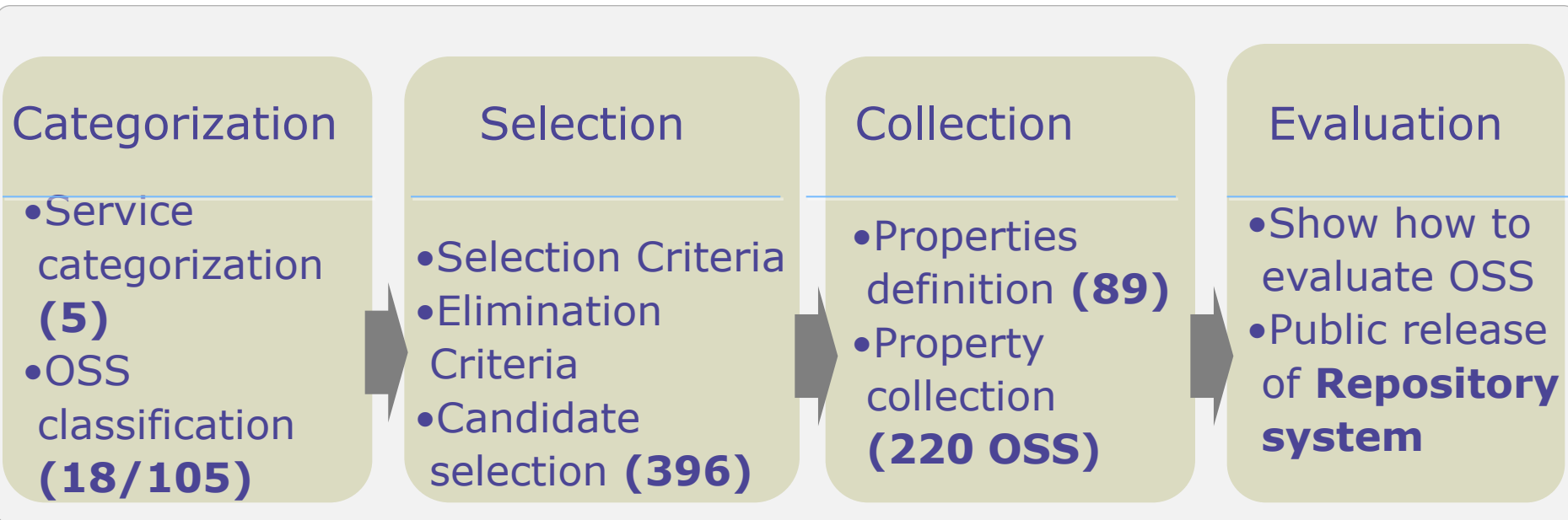
Own analyses



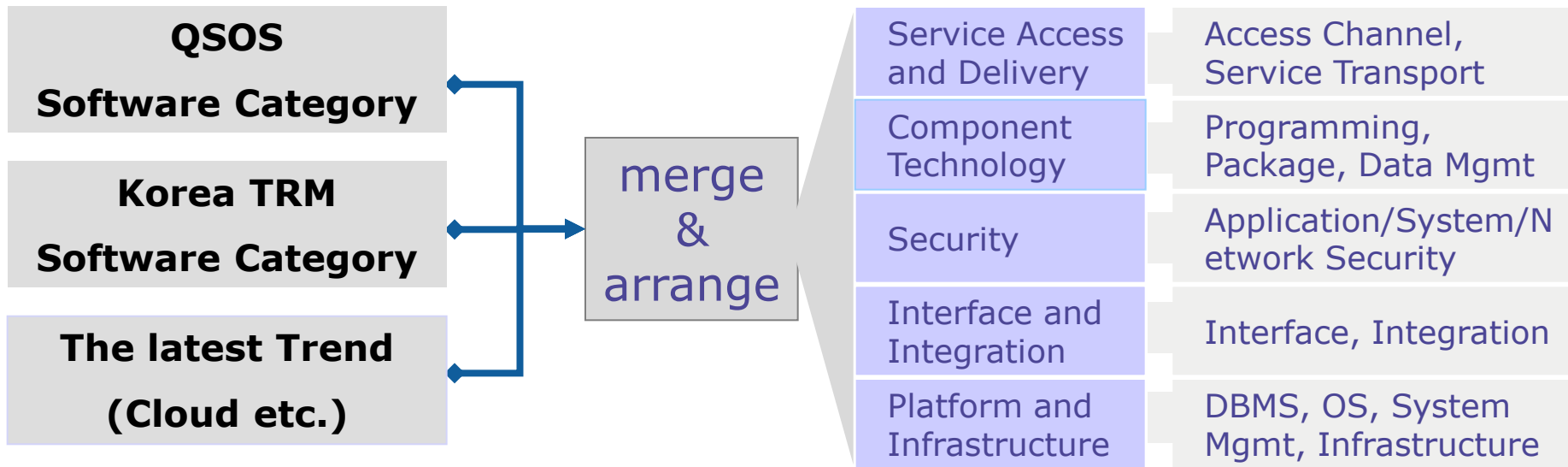
Flexible OSS Assessments

Steps

- To measure the quality of OSS by comparing and evaluating various properties



- Categorize 396 OSS projects based on the Korea government TRM (Technical Reference Model)
- 5 main categories, 18 sub categories, additional 105 sub-sub categories



- If too detailed, integrated into a higher category
- If too rough, divided into sub categories
- Functions for a specific business category are excluded

Collection

89 Properties Defined

1. OSS information must be retrieved from the **official and public websites**
2. **Quantitative data** which **can be evaluated** must be adopted
3. Detailed **Properties gathering Procedures**

License

Development System

History

Community

Specification

Composition

Quality

Recognition

Package

Business

Case Study

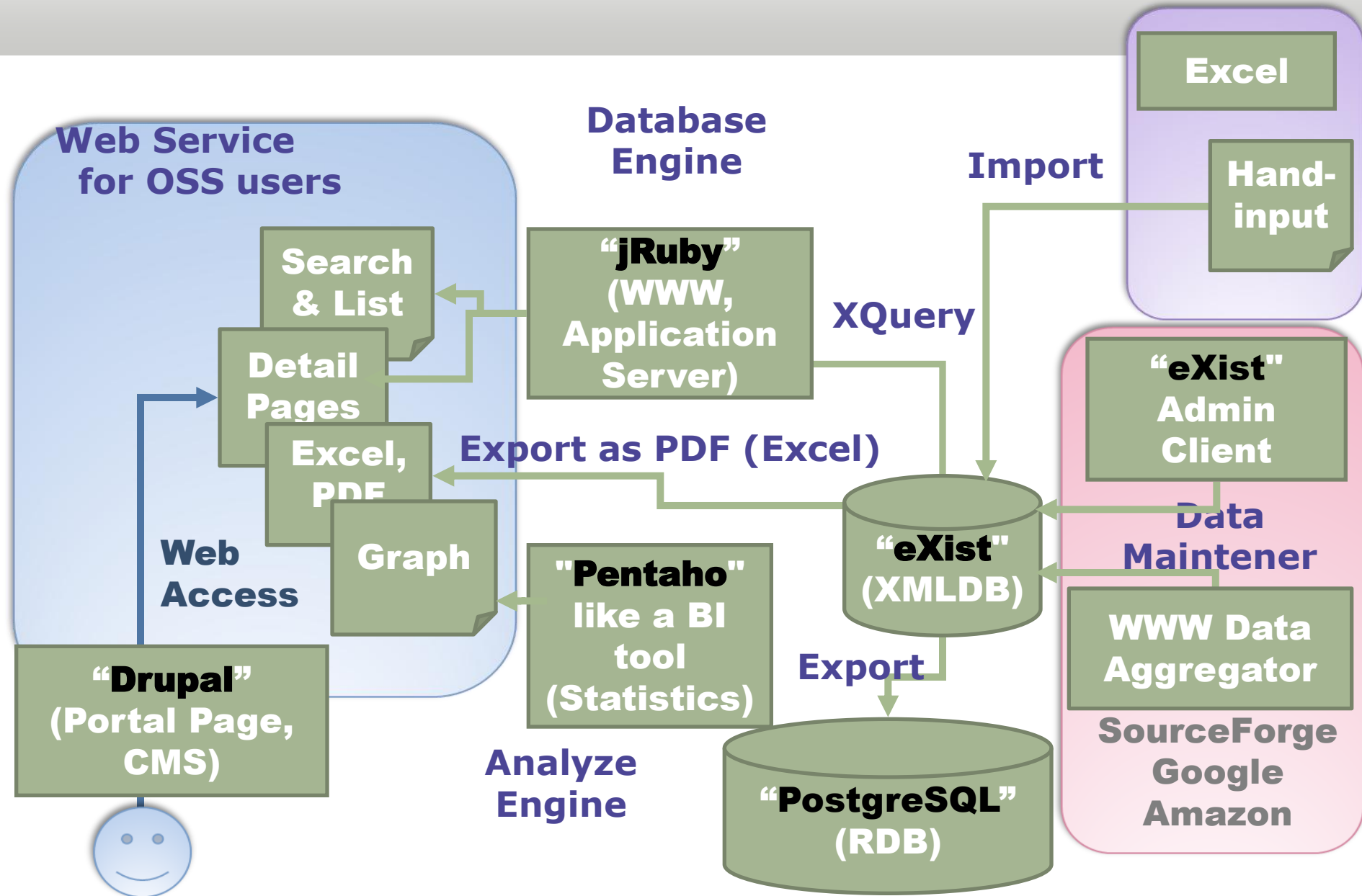
■ Easy to have access to OSS evaluation information:

- Properties and so on in RepOSS

■ Show how to evaluate OSS

- Mapping into existing OSS maturity assessment models (QSOS, OpenBRR, and so on)
- Analysis examples by BI tools

System Architecture



Implementation points

Easy to use the repository

- Data into XML database
- BI tools for analyses

Evaluation and Assessments

- OSS maturity model mappings
- Evaluation examples
- Data collection manual

Continuity and durability as a OSS community

- Public Open web system, which is being developed by **all OSS components**

OpenBRR Mapping



8.2 User contribution framework

1) Check it by using 4 "Communication Tools" including 8.3 "Developer Mailing List" and "User Mailing List"

9. Adoption

9.1 How many book titles does Amazon.com give for Power Search query: "subject computer and title: component name"?

1) 8.3.2 "Number of books"

9.2 Reference deployment

1) 11 "Case"

10. Community

10.1 Average volume of general mailing list in the last 6 months

1) Check it by using 4.3 "Developer Mailing List" and 4.4 "User Mailing List" (it's the same question as 7.1)

10.2 Number of unique code contributors in the last 6 months

1) 2.1 "Main Developer" and 2.2 "Other Developer"

11. Professionalism

11.1 Project Driver: N/A

1) 2 "Development System" will provide some hints

11.2 Difficulty to enter core developer team: N/A

2) Check it by checking 4.3 "Developer mailing list address"

Reference

[1] Business Readiness Rating for Open Source, BRR 2005-RFC 1,

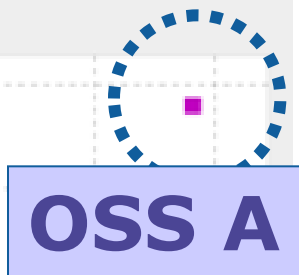
http://www.immagic.com/eLibrary/ARCHIVES/GENERAL/CMU_US/C050728W.pdf

Analysis with BI tools

Numbers of bugs

Correlation scatter chart

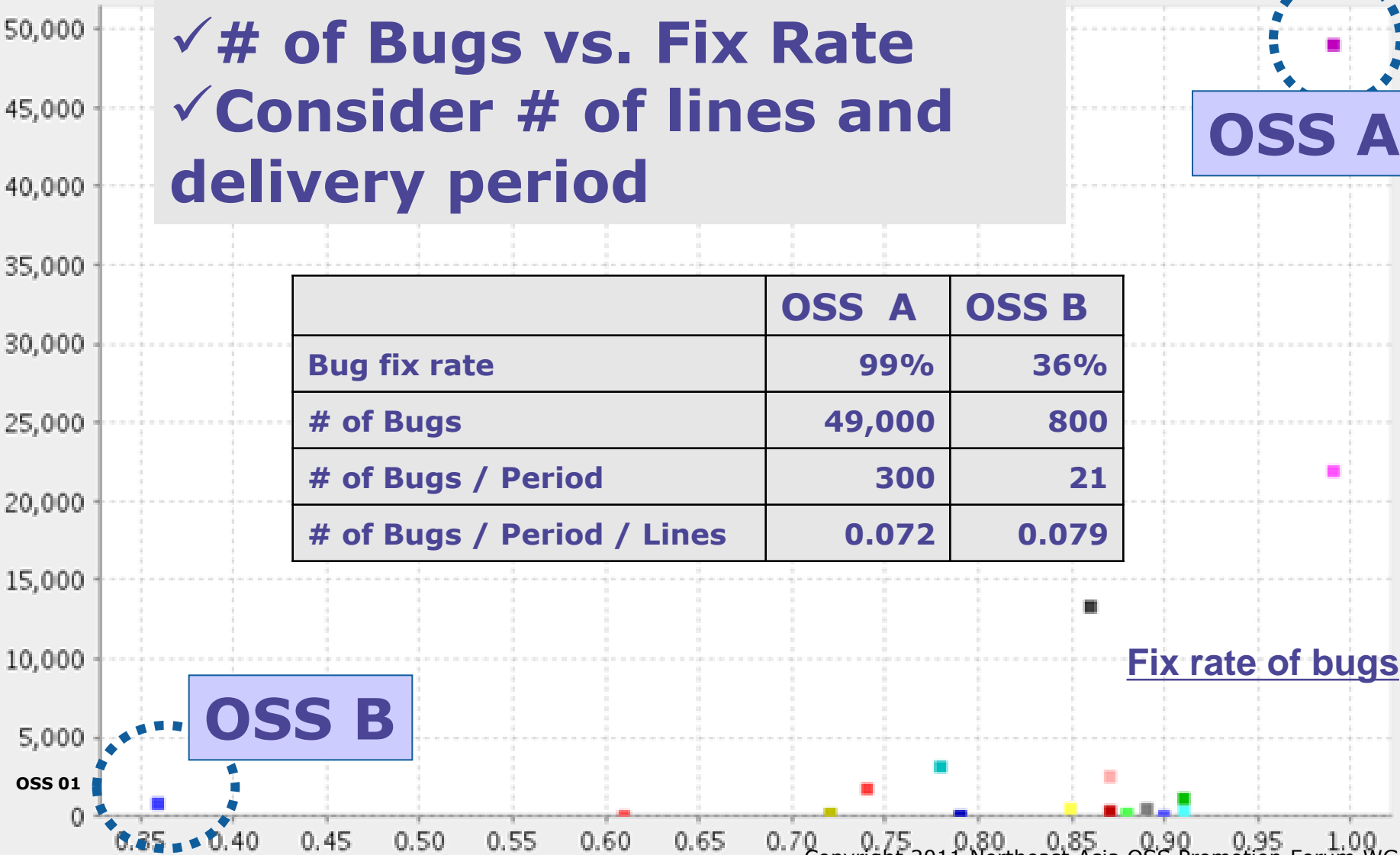
- ✓ # of Bugs vs. Fix Rate
- ✓ Consider # of lines and delivery period



	OSS A	OSS B
Bug fix rate	99%	36%
# of Bugs	49,000	800
# of Bugs / Period	300	21
# of Bugs / Period / Lines	0.072	0.079

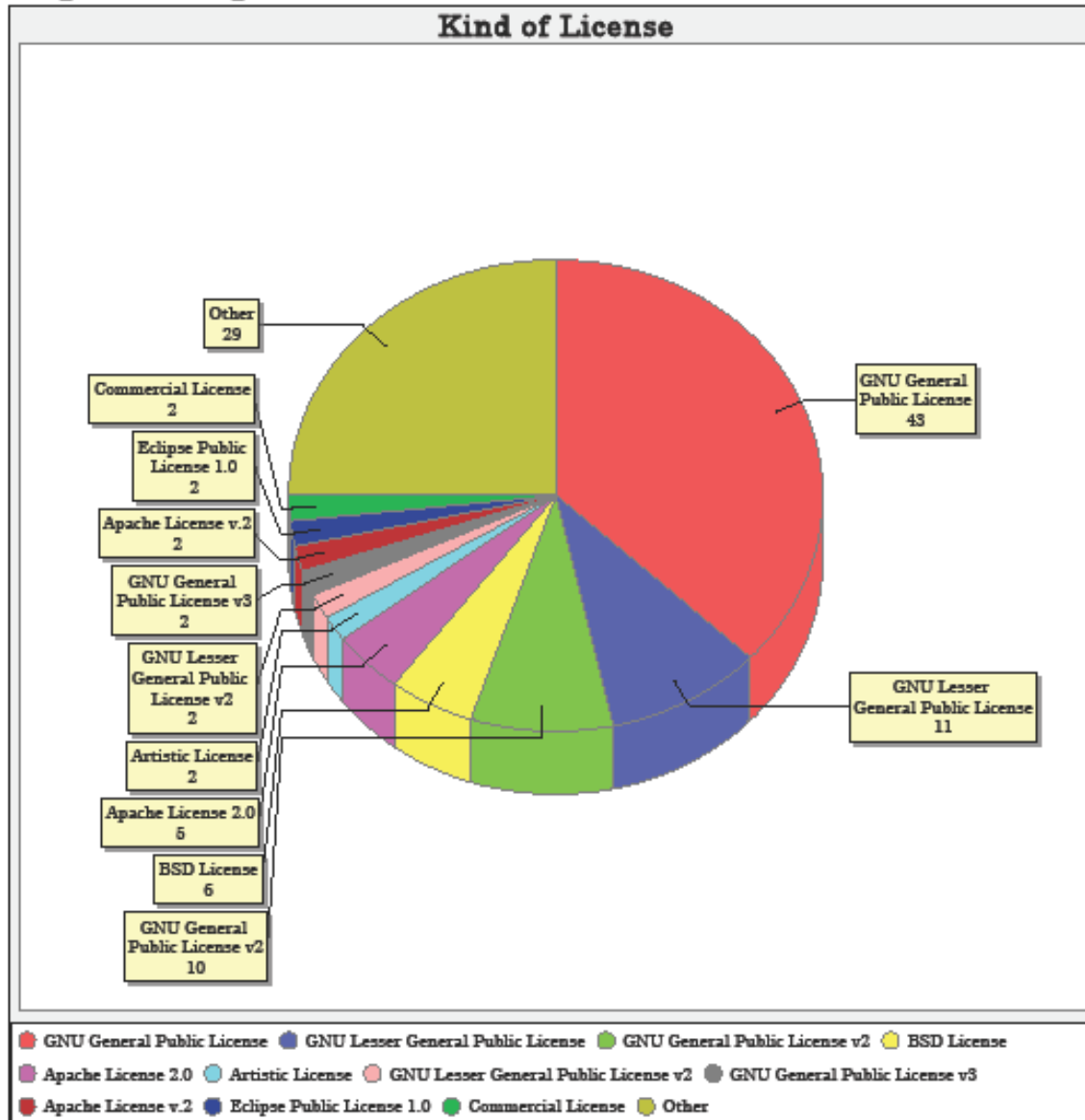


Fix rate of bugs



Licenses

RepOSS Report



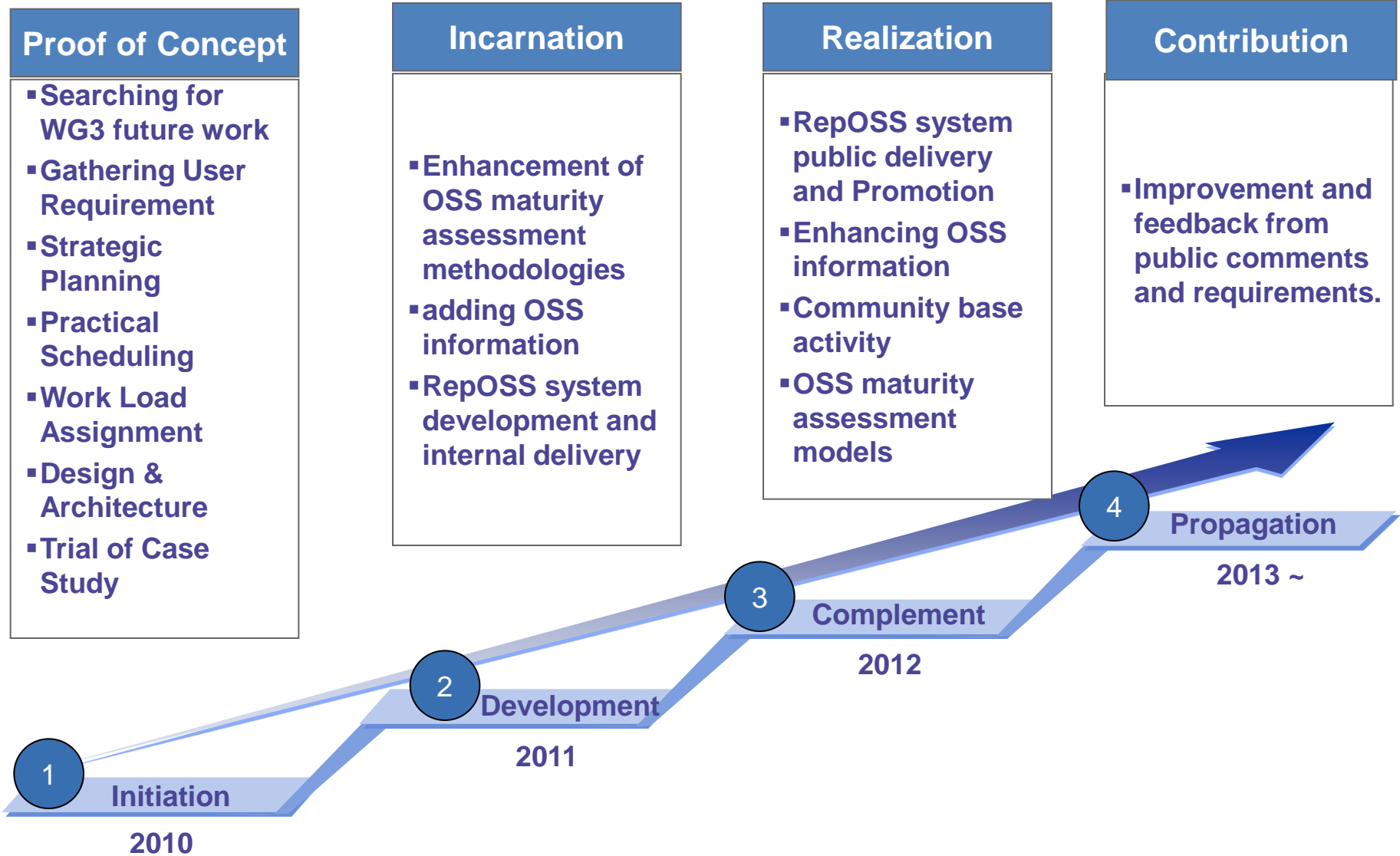
✓ Check OSS Licenses, whether or not they were approved by OSI (Open Source Initiative)

✓ What should be done under the license, which you want to use (i.e. "THE BEER-WARE LICENSE")

✓ Check the status like Dual or Triple licenses



RepOSS Road Map





Thank you

1. Deprez, J.-C., Alexandre, S., “Comparing Assessment Methodologies for Free/Open Source Software: OpenBRR & QSOS”
http://www.qualoss.org/dissemination/DEPREZ_CompareFLOSSAssessMethodo-Camera-02.pdf
2. Etiel Petrinja, Alberto Sillitti, and Giancarlo Succi, “Comparing Open BRR, QSOS, and OMM Assessment Models”
<http://www.inf.unibz.it/~gsucci/publications/images/ComparingOpenBRRQSOSandOMM%20Assessment%20Models.pdf>
3. George Kakarontzas, Panagiotis Katsaros, Ioannis Stamelos, “Component Certification as a Prerequisite for Widespread OSS Reuse”
<http://journal.ub.tu-berlin.de/index.php/eceasst/article/viewFile/449/433>
4. James Howison, Megan Conklin, Kevin Crowston, “FLOSSmole: A Collaborative repository for FLOSS research data and analyses”
<http://flosseb.floss.syr.edu/system/files/FLOSSmole.pdf>
5. Business Readiness Rating for Open Source BRR 2005- RFC 1 Method for Qualification and Selection of Open Source software (QSOS) v1.6 2006
<http://www.qualipso.org/>