How Open Source Communities to do Standardization

Li Li (Thomas), thomas.lili@huawei.com
VP of Standardization & Industry
For LinuxCon + ContainerCon + CloudOpen China, 19-20 June 2017, Beijing

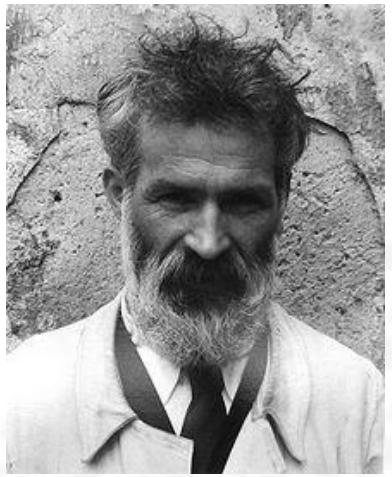
HUAWEI TECHNOLOGIES CO., LTD.

HUAWEI

Dilemma of Open Source

Create like a god.
Command like a king.
Work like a slave.

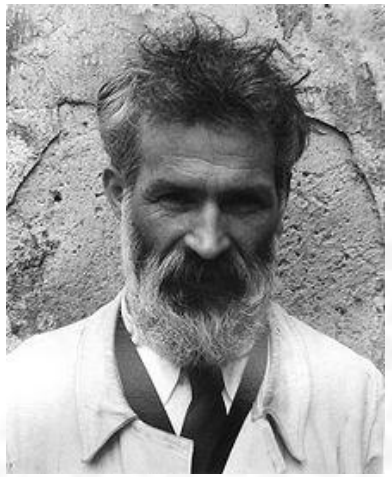
--- Constantin Brancusi

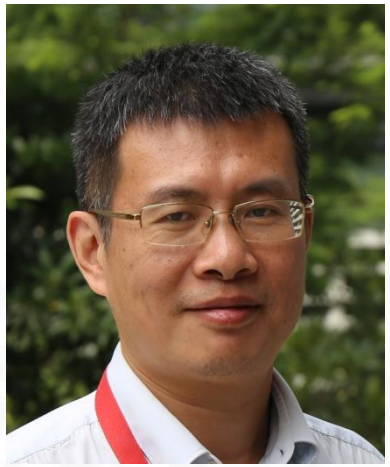


Dilemma of Open Source

Create like a god.
Command like a king.
Work like a slave.

--- Constantin Brancusi





But nobody is willing to be paid like a slave.
The open source code has huge potential.

--- Thomas Li

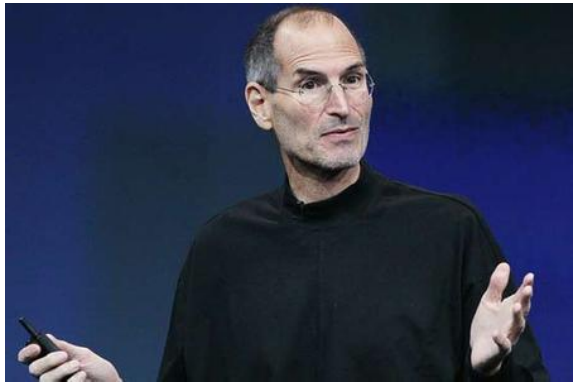
When Open Source meets Business

Open Source World	Business World
Culture: Technical Meritocracy	Culture: Business Success
Focus: Workable technologies	Focus: High quality products/services
Features: Open, Share and Free	Features: Differentiation, Profitability
Power source: Code talk	Power source: Customer talk

The right language to talk with customers and the industry is NOT code. The customers may not be capable to read codes or understand technologies, but they have the power to decide paying the bill or not.

Why Standards matter

Apple I
Presented as technology



Apple II
Presented as product

Standards define quality of products / services depending on customer requirements and technological feasibility. In most cases, standards IS the right language to talk with customers. It's an important factor for customers' decision-making.

Why Standards matter – Quality of products / services

The seven quality management principles are:

QMP 1 – Customer focus

QMP 2 – Leadership

QMP 3 – Engagement of people

QMP 4 – Process approach

QMP 5 – Improvement

QMP 6 – Evidence-based decision making

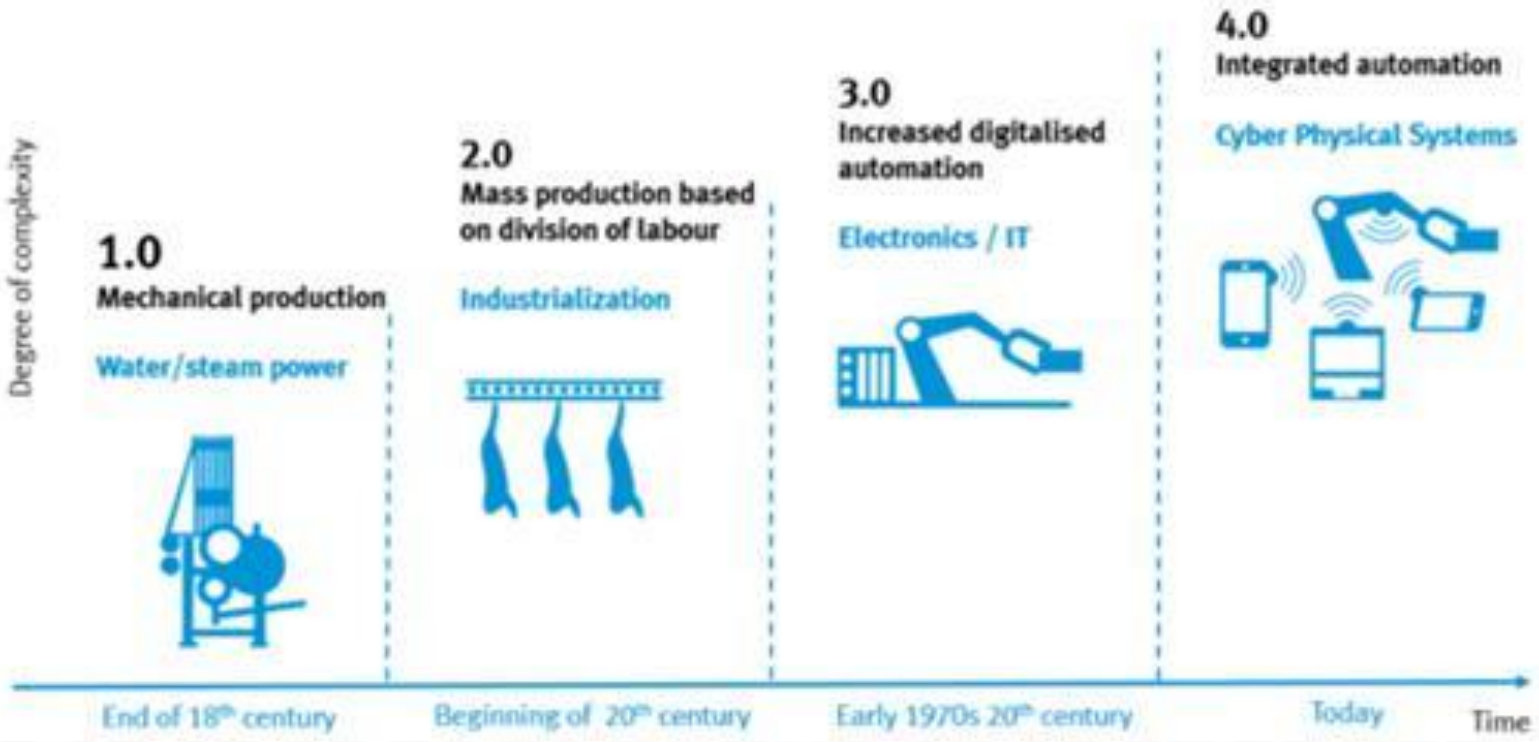
QMP 7 – Relationship management

Customers are willing to pay the bill for the products / services with certified quality.
The certification must be based on well adopted standards for fairness and neutrality.
It's crucial for high-tech products especially because of complexity.

Why Standards matter – Economy of scale / TCO

Very well standardized air interface facilitates the prosperous market of the mobile phone from 2G to 4G as well as 5G in the future. This is a perfect example of achieving economy of scale and lowering down TCO by means of standards. Standards is an important facilitator for massive deployment.

Why Standards matter – Market direction & roadmap



The customers make investment decisions not only depending on current situations but also future market direction and roadmap. Standardizing the future products in advance is a good way to decrease the uncertainties for decision-makers. Basically investment hates uncertainties.

Why Standards matter – Industry cadence & stability

IMT-2020 / 5G

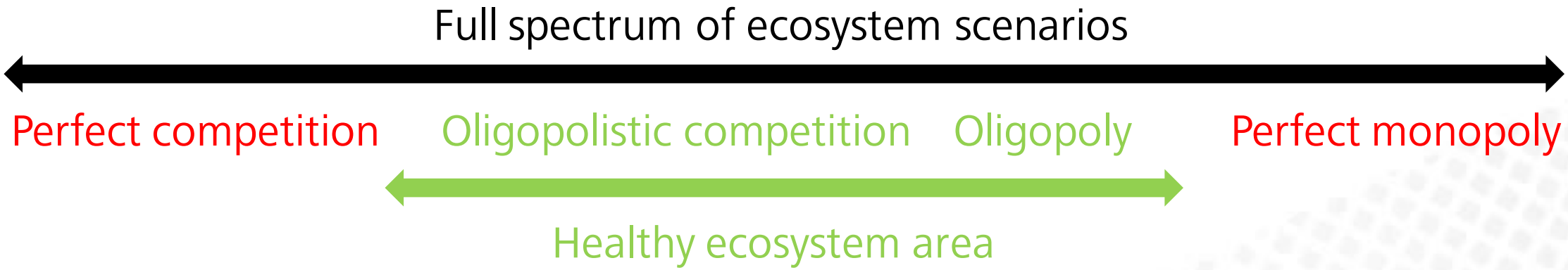
IMT-Advanced / 4G

IMT-2000 / 3G

2000s 2010s 2020s

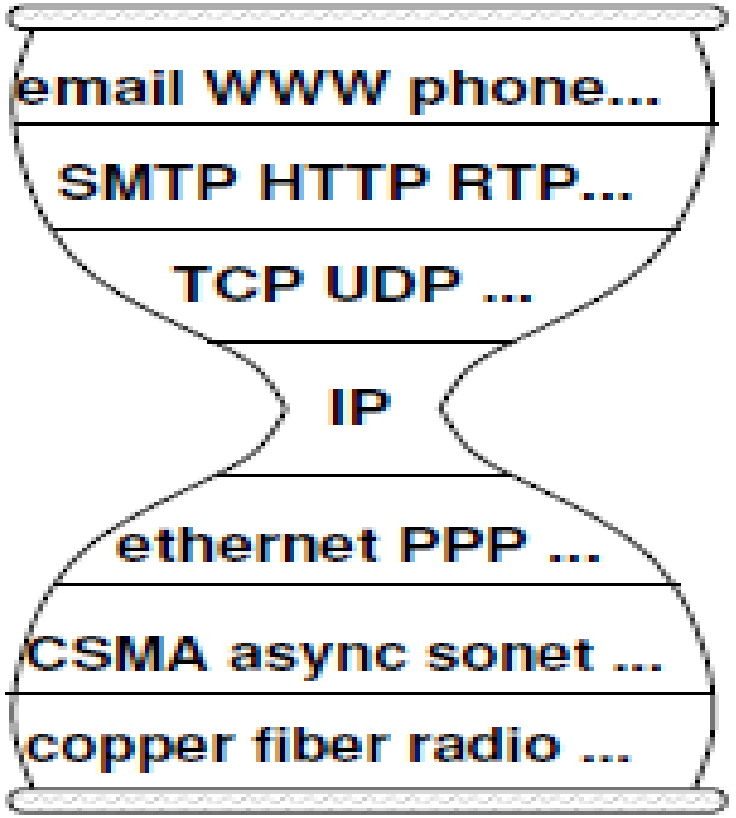
The ROI cadence of Telecom operators on heavy investment of mobile communication networks is about 7-8 years. The IMT standards family defined by ITU-R plays a crucial role of managing the industry cadence and protecting the network investment. Sometimes technology evolving too fast is not good for the industry.

Why Standards matter – Healthy ecosystem



Perfect competition makes everybody difficult to make profit. Perfect monopoly allows only one company making huge profit but leaves pain for customers and all other competitors. Balanced standardization can set a proper industry threshold of entrance to adjust the ecosystem falling in the relatively healthy area.

Why Standards matter – Backward compatibility



Internet and communication networks naturally have the requirements of inter-connection & backward compatibility. This makes the protocols (like IP) can't be revolutionarily re-written for a long time. Other similar scenario with strong legacy investment protection appeals would also have this requirement.

Standardization is necessarily to keep backward compatibility when new technologies introduced in a strong legacy investment protection scenario.

Why Standards matter – Regulatory requirements

European Commission

ETSI

Public safety
Cyber security
Health protection
Resource management
Sustainable development
etc.

中华人民共和国工业和信息化部
Ministry of Industry and Information Technology of the People's Republic of China

中国通信标准化协会
China Communications Standards Association

Anti-trust
Anti-fraud
Accessibility
Financial security
Lawful interception

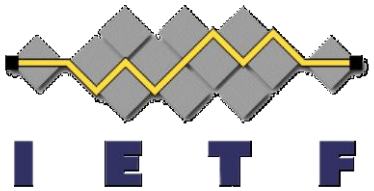
FCC Federal Communications Commission

atis

Energy efficiency
Privacy protection
Spectrum regulation
Environment protection
Electromagnetic compatibility

How Standardization works – Typical cases

Voting rule exists but rarely being used. Always trying to get full consensus with everybody happy. Gentleman's agreement sometimes is used to solve conflictions.



Never voting. Running code to ensure technical feasibility. Rough consensus.

Always voting to make decisions, depending on some extend of discussions. Following "Robert's Rules of Order" exactly.

How Standardization works – Principles

Step-1: Communication

Identify effect requirements of standardization.

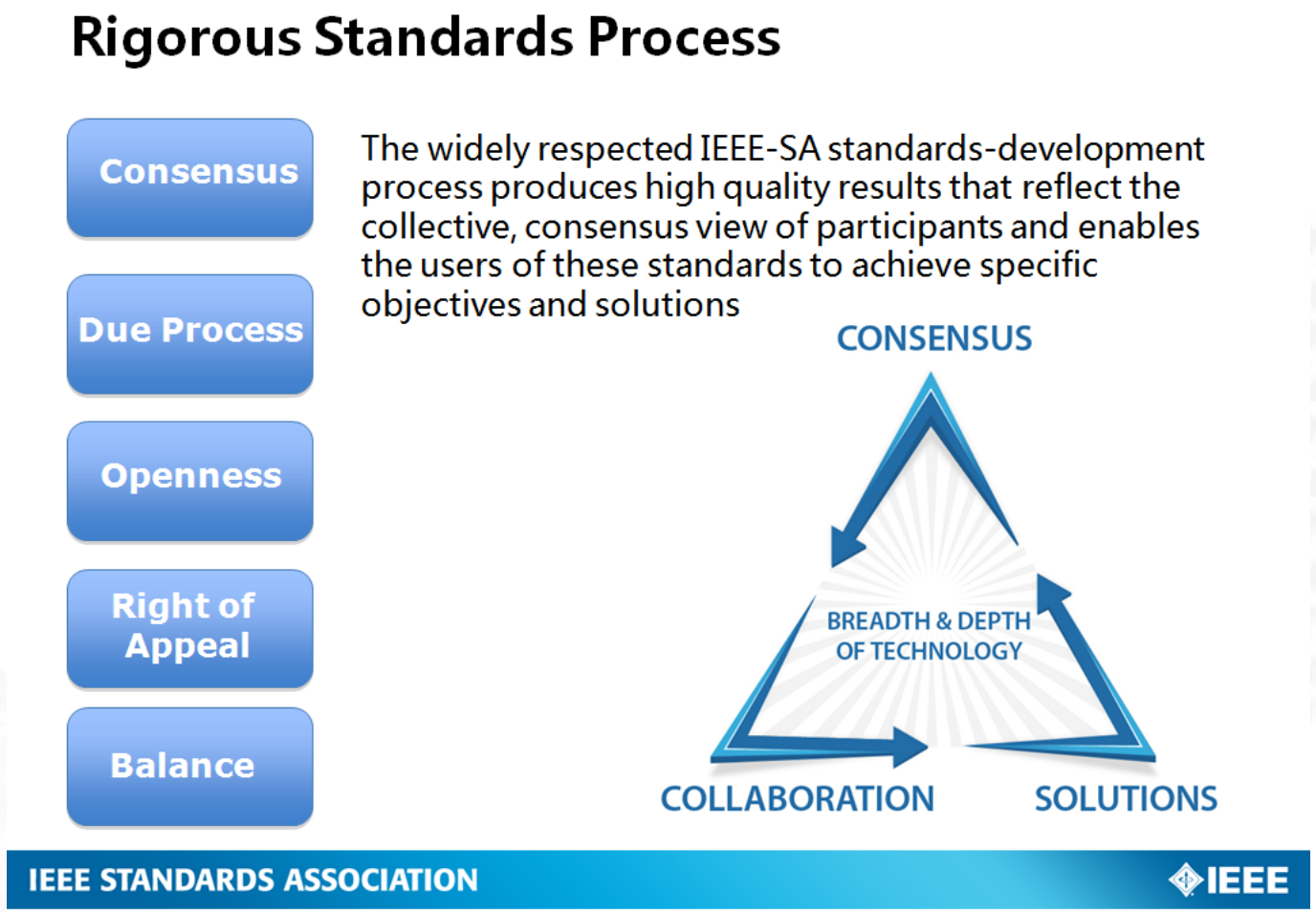
Step-2: Consensus

Collaborate and build consensus in a balanced way to create standards. Anti-domination.

Step-3: Commitment

Design, manufacture, test and acquire products / services against standards.

IEEE standards process as an example



Complains on some traditional standardization practices

Too inefficient!

10% people in the meeting room is working, another 10% is blocking, all others are following or sleeping.

Too bureaucratic!

They are actually lawyers pretending to be engineers. They focus on political and procedural issues much more than real business concerns.

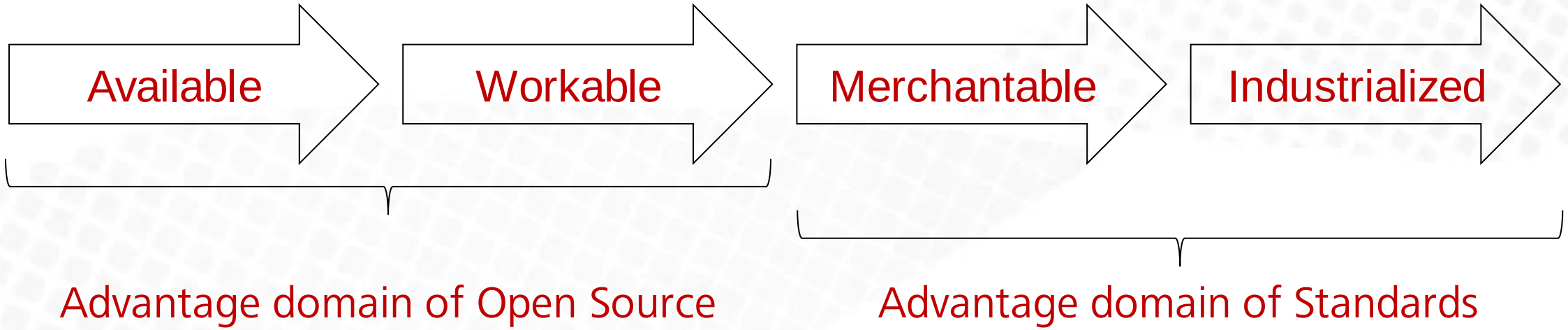
Too slow!

When the standards ratified finally, the market opportunities are already gone.

Do Open Source Communities really need Standards?

Yes, of course!

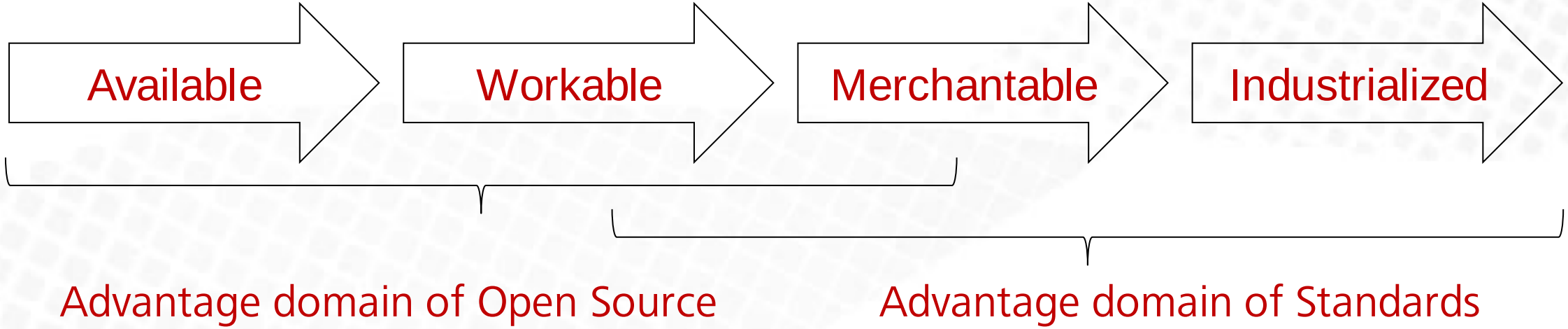
The open source code is free, but the products / services based on open source code need to get customers' buy-in.



Do Open Source Communities really need Standards?

Yes, of course!

The open source code is free, but the products / services based on open source code need to get customers' buy-in. Standardization is the right catalyzer to liberate the potential value of open source for the business world.



Challenge 1: Edge technologies vs. Un-clear requirements

While the programmers are busy on making code available, technology workable and fixing bugs…

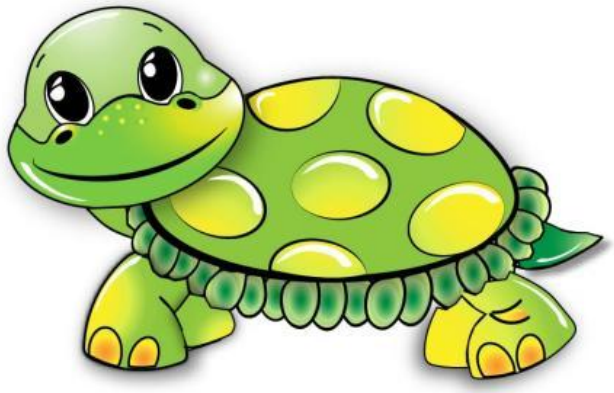
The customers are already urging on high-quality products.
But in some cases they don't have clear understanding of the technology and confirmed requirements.

Practical advice: Don't be in hurry and coordinate more.

- Set a "User Advisory Group" in the open source community to engage customers.
- Start from technology whitepapers in the community for preparation on following serious standardization.
- Do some trials with customers to find out real requirements and technical gaps.

Challenge 2: Fast Innovation vs. Industry cadence control

While the innovations are under quick iteration in the open source community…



The target industry is still in slow motion on new technology adoption.
The decision-makers are wondering their wise decision would become a stupid one just after 3 months later.

Practical advice: Try to understand the industry logic and don't break it easily.

- Quick tech-iteration means high expense on R&D and threaten of legacy investment.
- Fast change of technologies is an uncertainty for business decision-making.
- Finding out bottlenecks of the target industry is more important than creating fancy technologies.

Challenge 3: Coding vs. Debating

While open source programming needs in-depth skills for creating beautiful codes…

Standardization needs compound debating skills including language, negotiation, strategy, politics, technology, market, investment, psychology, and even sense of humor.
Yes, it's a lot.

Practical advice: Be confident. Programmers CAN be excellent standards experts.

- Open mind, flexibility, fast learning, collaboration are common features of open source and standardization.
- Only real technical experts can create good standards. Pure politicians and lawyers can't.

Challenge 4: Stability vs. DevOps

While standards should be stable enough to be cornerstones of the industry…

The DevOps methodology widely adopted in open source world will make the code to be a moving target. People would wonder there is no standardization need because DevOps can fulfill requirements dynamically.

Practical advice: Focus on standardization of the architecture and key interfaces.

- No matter how dynamic the code is, some key interfaces must be stable for a prosperous ecosystem.
- Smooth DevOps is always based on a well-designed, stable software architecture and strong hardware support.

Principle 1: Customer-Centric

- Identify who the customers are
- Identify what the requirements are
- Engage the customers in early stage of open source coding to dynamically adjust towards the right direction
- Start standardization against open source-based products / services in proper time synchronizing with industry pace

The most valuable contribution of the customers in open source communities is not code, but requirements and commitment of buy-in.

Principle 2: Don’ t invent wheel

- Mobile communication network standards
- Internet connection protocol standards
- LAN/MAN standards
- Container related standards

Duplicated standardization efforts in different standards bodies will cause fragmentation of the industry. Successful standards always rely on real technical experts in the specified areas.

Principle 3: No Over-Standardization

- Chinese ancient idiom: Paint a snake with feet
- Over-standardization is a poison of innovation.
- Typical case: Code is standard.
- The balance between standardization and innovation should be considered.

For software, architecture design is crucial. It could be necessarily to be standardized. Some interfaces should be standardized, some other interfaces maybe unnecessary. Some data models / information models and procedures may need standardization. But most parts of software don't need.
It's always case-by-case. We need to scope the standards carefully every time.

Principle 4: Be Patient

- Standardization is a multilateral roundtable process. It's supposed be in democracy mode.
- People can write 100 pages overnight in meritocracy mode, but this is not standardization.
- Doing standardization in open source communities, the balance between meritocracy and democracy is a key for succeed.

The bad practices in traditional standardization like too inefficient, too bureaucracy and too slow should be avoided. But we still need to be patient because standardization's nature is not simplicity. A new standardization style needs to be created inside open source communities.

Standardization possibilities in open source world

CVP needs standards. Collaborating with ETSI ISG NFV, 3GPP, TMF and/or OPNFV doing by itself would be options.

NBI standard is important to build APP ecosystem. DefCore could handle it.

Blockchain needs standards to facilitate the industry adoption.

Existing storage SDO could be collaborated. New standards would be necessary inside OpenSDS.

Collaborating with TMF, MEF, IETF, OASIS, BBF etc. is important.

Open source security is a fundamental infrastructure. Standards are necessary finally for commercial certification.

Thank you

www.huawei.com