

# Through the Looking Glass of Immaterial Labor

Yunwen Ye (Software Research Associates Inc.)

Kumiyo Nakakoji (Software Research Associates Inc.)

Yasuhiro Yamamoto (Tokyo Institute of Technology)

Kouichi Kishida (Software Research Associates Inc.)

## *Message*

# De-“Software Engineering”

- There is no future in *Software Engineering* shaped by the Taylorist criteria
  - Serialization of work
  - Parcelization of work
  - Insignificance of individual workers
- But, software development is setting the trends
  - Mode of software production is becoming the dominant mode of industrial production

*Maurizio Lazzarato, 1996*

# Immaterial Labor

- A philosophical framework to understand and analyze post-Taylorist production
  - Immaterial labor is the new dominant mode of production
- Produce the informational and cultural contents of a commodity
  - Create services, knowledge, communication
  - Involve skills of cybernetics and computer control
  - Define and fix cultural and artistic standards, fashions, tastes, consumer norms, and public opinions
  - Satisfy the affective needs of users: feeling, passion, satisfaction, excitement, and sense of connectedness

## *Creation of Demands*

# Production, Distribution, Consumption

- Taylorist production
  - Mass production of standardized commodities
  - Innovation lies in the rationalization of labor
- Toyotatism production
  - Production after having orders
  - Innovation lies in the distribution system
- Immaterial Production
  - Production of new subjectivity and social reality
  - Innovation lies in the creation of new demands

## *What Went Wrong*

# Value and Quality of Software

- Software development is a typical example of immaterial labor
- We have focused narrowly on functionality (the objective sphere)
- Three spheres of software value and quality
  - Objective sphere
    - by prediction and control
  - Social sphere
    - through communication and interpretation
  - Subjective sphere
    - aesthetic and emotional experience

*Objective, Social & Subjective Spheres of Software*

# Producing Software

- To use is to produce
  - Values in social and subjective spheres are determined by the use of software
  - There is no correct system to be built
- To produce is to generate new needs and requirements
  - Enrich and alter technological, social and cultural “realities” of users
  - Ever-changing requirements are the beacons of success

*Software is not only a typical example of immaterial labor*

# Are we looking backward ?

- *Software Engineering* is the wrong metaphor
  - Taylorist production is the obsolete model to look up to
- Software is the enabler of immaterial labor
  - Move workers to the side of production
  - Informationalization of industry
  - Transforming material labor to immaterial labor

## *Issues & Challenges*

# Organizing Software Development

- Conceptualization and execution is inseparable
- Boundary of leisure & labor blurs
- Managed as individuals, rather than roles
- Developers are not objects of control but subjects of communication
- Management, structuring and activation of “subjective” social collaboration
- From division of labor to distribution of knowledge



# De-“Software Engineering”

- There is no future in *Software Engineering* shaped by the Taylorist criteria
  - Serialization of work
  - Parcelization of work
  - Insignificance of individual workers
- But, software development is setting the trends
  - Mode of software production is becoming the dominant mode of industrial production

Don't Panic.