



SE463

Software Requirements Specification & Analysis

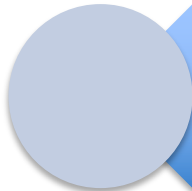
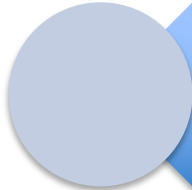
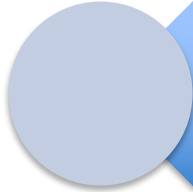
Stakeholders

Readings:

Karl E. Wieggers and Joy Beatty. *Software Requirements*, 3ed.
Microsoft Press, 2013.

- Chapter 2: “Requirements from the customer's perspective”
- Chapter 6: “Finding the voice of the user”

Module Objectives

-  Identify stakeholders
-  Consider multiple user classes
-  Use Personas to understand the needs of an absent user

Stakeholders

- A **stakeholder** is anyone who has a **stake** in the ultimate success of the project.
- During problem identification, we want to consider **all** of the stakeholders.
 - But sometimes we have to make do with **proxies**



© 2007 Oxford Creativity

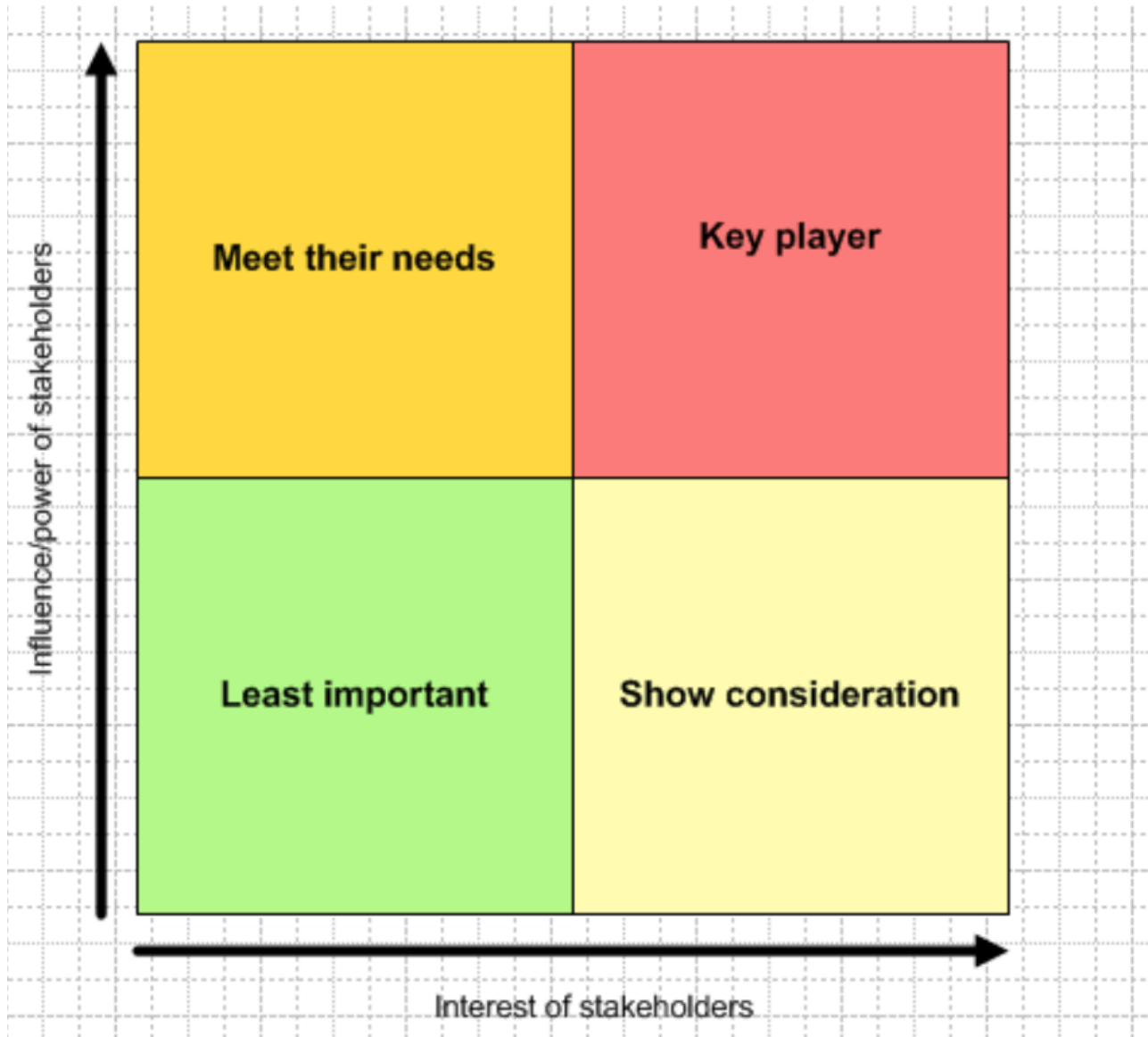
Importance of Stakeholders

They are important because:

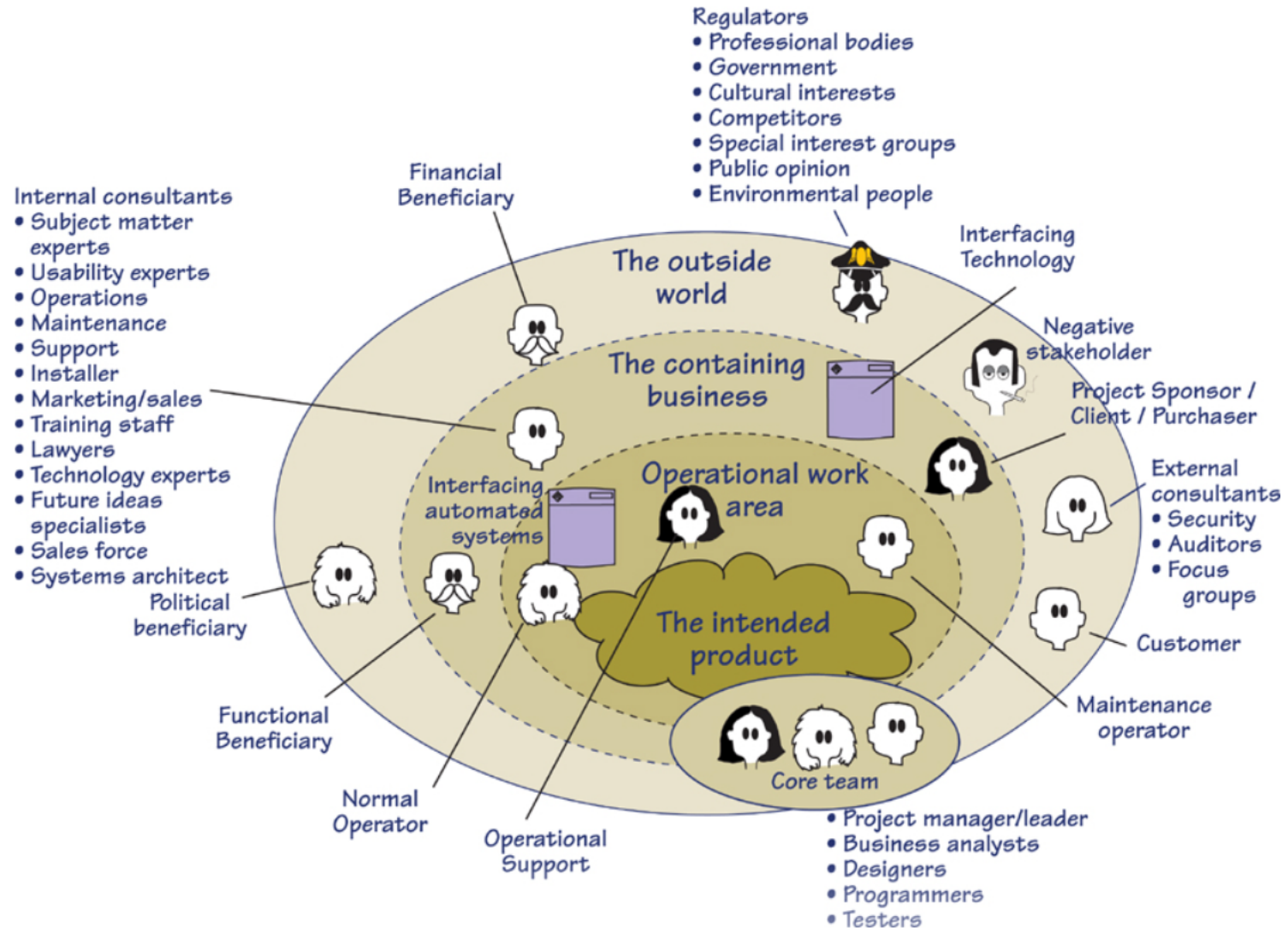
- They can speed up, slow down or completely obstruct our progress
- Making them happy and meeting their expectations will reduce the risk of negative influences
- We can improve the quality of the results we deliver.
- Identifying them is a continuous process during the entire project life cycle.

Remember, all stakeholders are not alike, they can have different needs and expectations. Thus, we have to treat them individually according to their needs.

Stakeholder Matrix



Stakeholders



Robertson, Robertson, *Mastering the Requirements Process*, 3rd ed, 2012.

Stakeholders: Owner/Client/Champion

- The *owner/client* is the person paying for the software to be developed
 - They are the ultimate stakeholder — almost always, the client has the last say in what the product does
 - For “bespoke” or customized systems, the client is the person with the chequebook 😊
 - For software for the mass market, the client may be the company developing the software
 - For in-house software, the client is probably the manager of the product’s users
 - Since his/her employees will be the primary beneficiaries, it is reasonable for him/her to pay for the project
 - Likely represented by a proxy

Stakeholders: Customer

- *A customer is a person who buys software after it is developed; possibly a manager*
 - May be the same as the client
 - May be the same as the user; other times, the customer is an office manager who buys software for his / her staff
 - For what requirements will he / she pay? Which are trivial or are excessive?
 - Must be an active participant in the project (or have an active representative if there are many customers)
 - Sometimes a marketing person or experienced manager serves as a proxy

Stakeholders: Users

- *Users (of both the current and future systems)*
 - Experts on the Work being performed
 - Experts on the existing system (if one exists) — tell us which features to keep and which need improvements, or
 - Experts on competitors' products — give suggestions about how to build a superior product.
 - May have special needs or requirements to be satisfied, e.g., regarding usability, desired features.

Stakeholders: Domain expert

- *Domain experts* know the problem domain well
 - Familiar with the problem that the software must solve.
e.g., financial experts for financial packages,
aeronautical engineers for aircraft navigation systems,
meteorologists for software that models the weather,
travel agents for travel industry, etc.
 - ... and are familiar with typical users and their expectations
 - ... and are familiar with typical deployment environments

Stakeholder: Software engineer

- *Software engineer* == technology expert
 - May include managers too
 - Represents the rest of the development team (developers, testers)
 - Ensures that the project is technically and economically feasible
 - Accurately estimates the cost and development time of the product
 - Educates the customer about innovative hardware or software technologies, and recommends new functionality that takes advantage of these technologies

Other Stakeholders

- *Inspectors == experts on government and safety regulations*
 - Familiar with government and safety regulations relevant to project
e.g., safety inspectors, auditors, certification inspectors
- *Market researchers (distant proxy for own customer interviews)*
 - Experts who have conducted surveys to determine future trends and potential customers' needs.
 - May assume the role of client, if the software is being developed for the mass market and there is no identifiable customer.
- *Lawyers*
 - Familiar with legal requirements
 - Familiar with licensing, e.g., for use of open source components

Other Stakeholders

- *Experts on adjacent systems*
 - They know about the interfaces to adjacent systems, and any special demands for interfacing with the adjacent system.
 - They need to be forewarned about the planned new system -- they could be affected by your work!
 - They can help to improve your work, by enhancing the services they provide.
- *Negative stakeholder*
 - It is best to understand why they do not want the project to succeed
 - They may simply have competing requirements that you'd be best off considering

Other stakeholders

- Consultants
- Product Management
- Subject-Matter Experts
- Core Team
- Public Opinion
- Government
- Cultural Interests
- Technical Experts
- Value-adders
- Experts on adjacent systems
- Negative stakeholder

Multiple Stakeholders

Different stakeholders may have different ideas about what the software should do.

Example: Student Academic Records

- Who are the stakeholders?
- What information does each stakeholder might like?

User Classes

Users can be categorized by their differences:

- Their access privilege or security levels (such as ordinary user, guest user, administrator)
- The tasks they perform during their business operations
- The features they use
- The frequency with which they use the product
- Their application domain experience and computer systems expertise
- The platforms they will be using (desktop PCs, laptop PCs, tablets, smartphones, specialized devices)
- Their native language
- Whether they will interact with the system directly or indirectly
- ➔ Disfavored users (who should not have access for legal, security, or safety reasons)
- ➔ Software agents, or bots

Personas



“A **Persona** is a user archetype we can use to help guide decisions about product features, navigation, interactions and even visual design.”

- Kim Goodwin, Cooper

Personas are a representative behavior and activity profile for a customer base. They are specific to the particular application or service.

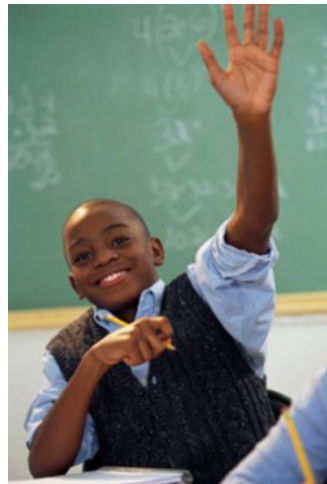
Personas



Personas are useful when real users are not available or are too numerous to interview them all.

- Important class of user, with unique needs and characteristics
- Include enough details (including a name!) to make the persona seem real to the team

Ken
(the keener)



Dudley
(the distracted)



Where does the data come from?

We derive our persona from multiple data sources such as:

- key stakeholders, web analytics, and customer support (Internal sources)
- The customer or someone we know (External sources)

Also, we can conduct **market research** or other **surveys**.

The degree of precision needed varies in proportion to both the size of the user base and the criticality of the product to be built.

How to write personas

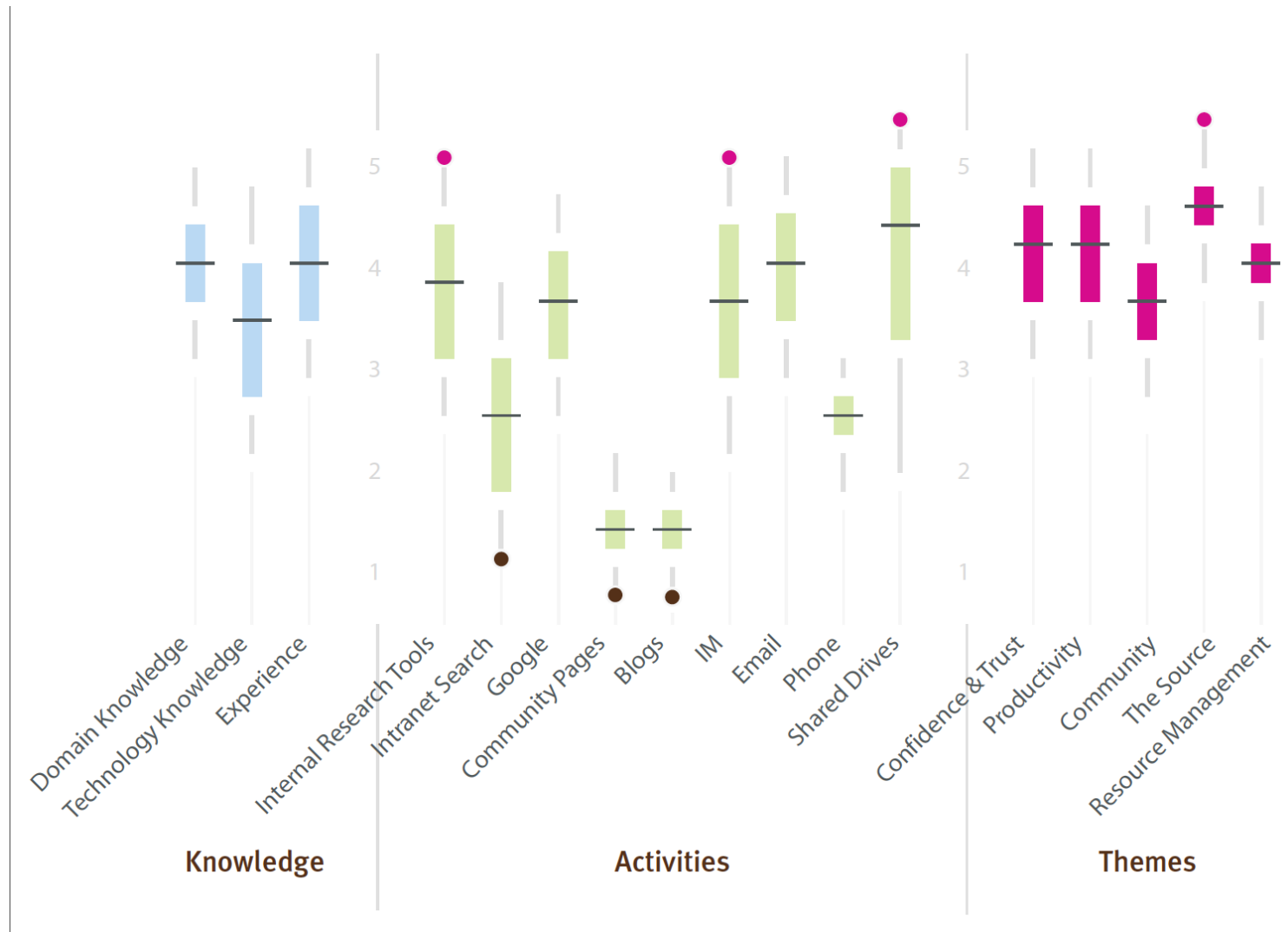
You want to identify trends or patterns in user behaviors, expectations and motivations.

How?

- The best way is to interview real users. Think of current or potential user.
- Interview business stakeholders that interact frequently with users.
- Review market research.
- Survey users and business stakeholders.
- Talk to friends and family that are users of your competitors.

[illegible]

Data-Driven Personas



© Todd Warfel 2007. *Data-Driven Personas*. <https://www.slideshare.net/toddwarfel/data-driven-personas/31>

Data-Driven Personas

The Moderately Seasoned Professional

Michael



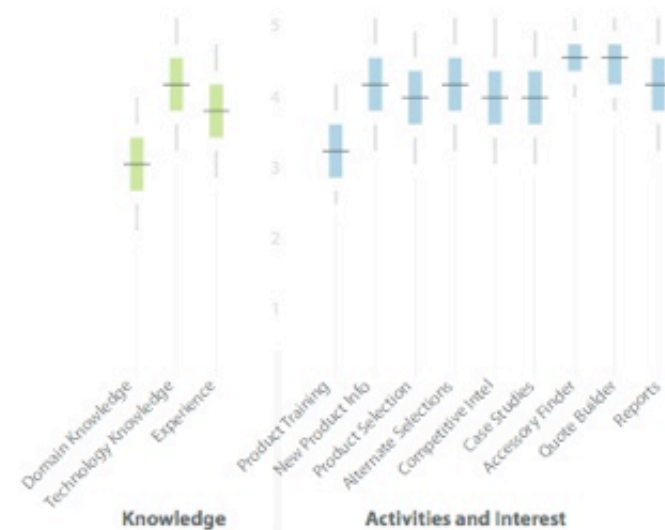
“I’d like to see a good, better, best.”

MICHAEL HAS BEEN IN SALES for YEARS AND HAS BEEN SELLING AND OTHER products for most of them. he's pretty comfortable with the Symbol products and isn't that interested in basic product info, but finds himself wondering if there's a better alternative than the product he's suggesting. It's a challenge keeping up to date on all the product info from Symbol and the other lines he sells. he'd love to see something that recommends a good, better, and best option when they're available as well as showing him where the particular model stacks up against other competitive products.

The right tool for Michael helps him pick the best product while recommending other alternatives. It also has case studies with examples of how other more seasoned salespeople have been able to upsell in similar environments and applications. Accessories and add-on services for Symbol scanners are a must. And if it is a frontend for Solution Builder that would be a huge bonus.

AGE: 42

OCCUPATION: regional Sales



Primary Use

- Case studies
- Alternate product selections (good, better, best)
- Accessory finder
- Product info for new products or product refresh
- As a frontend to Solution Builder

Goals

- Pick the right product and find better alternatives.
- Keep up-to-date on competitive intelligence.
- Increase sales volume.
- Increase accessory and add-on sales.
- Close more deals faster.

Influencers

- Easy-to-use
- Speed
- New product info
- Ability to run reports on open vs. closed quotes
- Ability to generate quotes

Frustrations & Pain Points

- having multiple usernames and passwords, he already has several for the different applications he uses at work and email – he doesn't need another one.
- A tool that leaves him in the dark

© Todd Warfel 2007. *Data-Driven Personas*. <https://www.slideshare.net/toddwarfel/data-driven-personas>