
User Stories Applied

For Agile Software Development

XP Atlanta

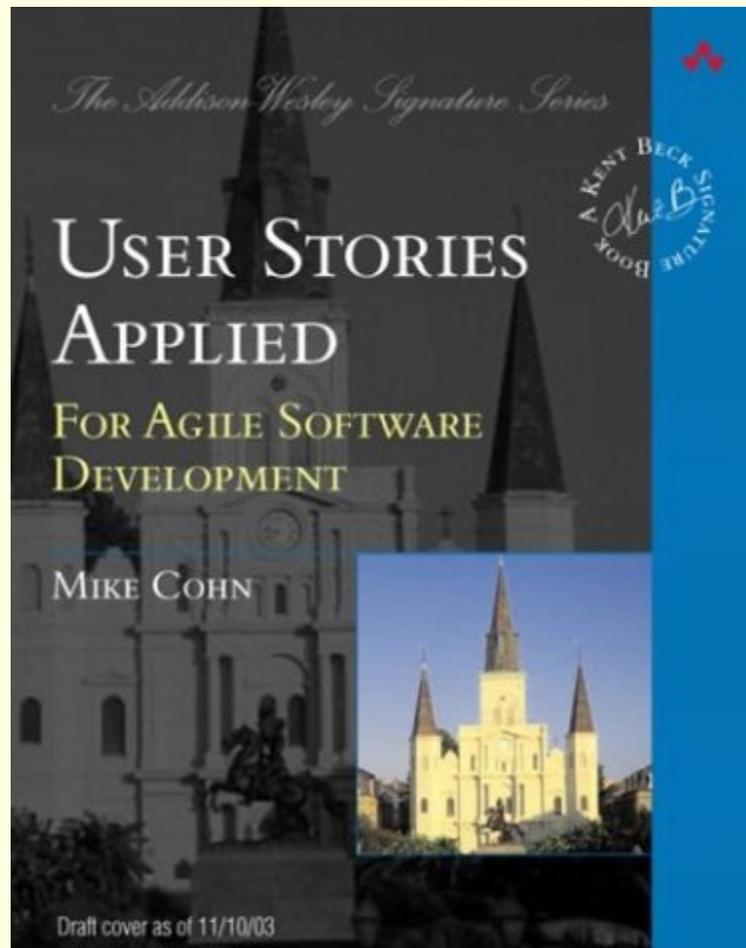
February 10, 2004

By Mike Cohn



Mountain Goat
S O F T W A R E

My books and background



- Programming for 20 years
- Past consulting to Viacom, Procter & Gamble, NBC, United Nations, Citibank, other smaller companies
- Founding member and director of the Agile Alliance
- Currently VP of Engineering with Fast401k in Denver, CO

Today's agenda

- What are user stories?
- Why user stories?
- User role modeling
- INVEST in good stories
- Guidelines for writing good stories

Ron Jeffries' Three Cs

Card

- Stories are traditionally written on note cards.
- Cards may be annotated with estimates, notes, etc.

Conversation

- Details behind the story come out during conversation with customer

Confirmation

- Acceptance tests confirm the story was coded correctly



Samples – Travel Reservation System

A user can make a hotel reservation.

Users can see photos of the hotels.

A user can cancel a reservation.

Users can restrict searches so they only see hotels with available rooms.

Where are the details?

- A user can make a hotel reservation.
 - Does she have to enter a credit card?
 - If so, what cards are accepted?
 - Is the charge applied immediately?
 - How can the user search for the hotel?
 - Can she search by city?
 - By quality rating?
 - By price range?
 - By type of room?
 - What information is shown for each room?
 - Can users make special requests, such as for a crib?

Details added in smaller “sub-”stories

A user can make a hotel reservation.

A user can search for a hotel. Search fields include city, price range and availability.

A user can view detailed information about a hotel.

A room can be reserved with a credit card.

Details added as tests

- Tests are written on the back of a story card
 - Can be used to express additional details and expectations

A user can make a hotel reservation.

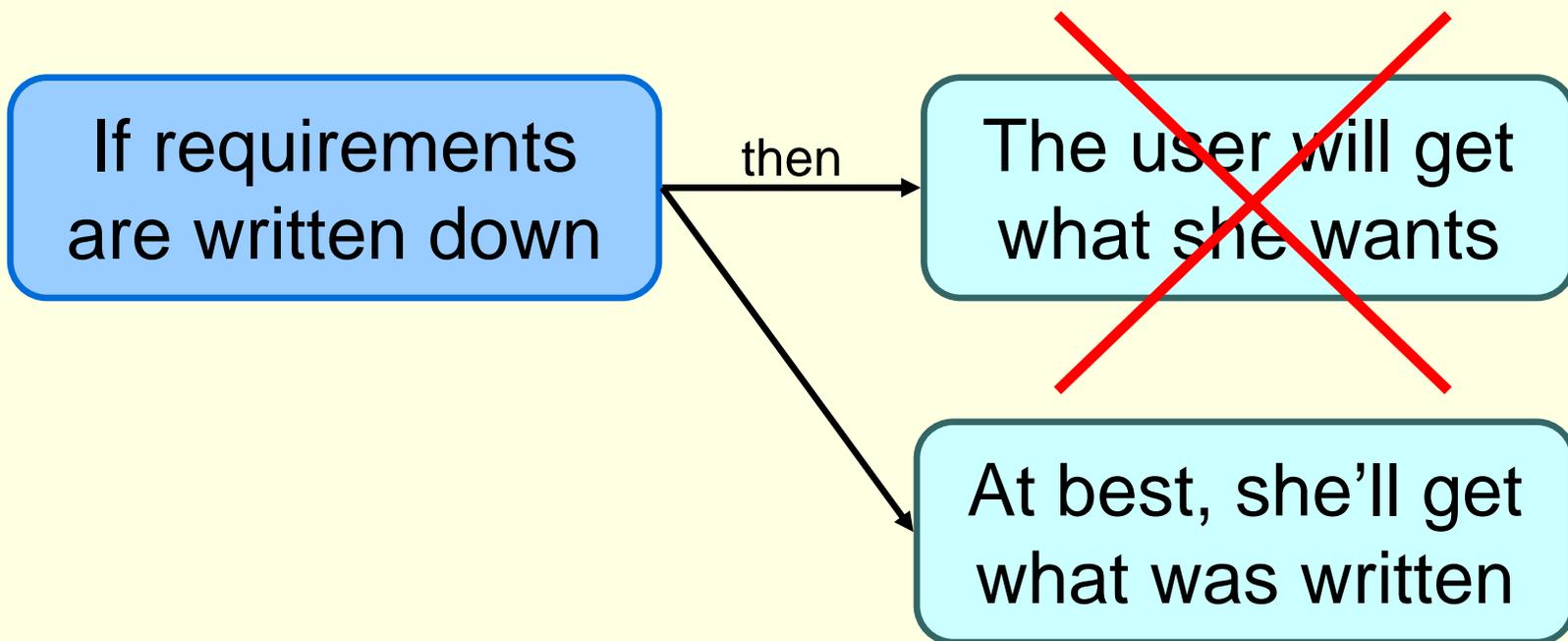
- Try it with a valid Visa then a valid MasterCard.
- Enter card numbers that are missing a digit, have an extra digit and have two transposed digits.
- Try it with a card with a valid number but that has been cancelled.
- Try it with a card expiration date in the past.

Today's agenda

- What are user stories?
- Why user stories?
- User role modeling
- INVEST in good stories
- Guidelines for writing good stories

So, why user stories?

- Shift focus from writing to talking



- “You built what I asked for, but it’s not what I need.”

Words are imprecise

Entrée comes
with soup or salad
and bread.

- (Soup or Salad) and Bread
- (Soup) or (Salad and Bread)

Actual examples

The user can enter a name. It can be 127 characters.

- Must the user enter a name?
- Can it be other than 127 chars?

The system should prominently display a warning message whenever the user enters invalid data.

- What does *should* mean?
- What does *prominently display* mean?
- Is *invalid data* defined elsewhere?

Words have multiple meanings

Buffalo buffalo buffalo.

- Bison intimidate bison.

Buffalo buffalo Buffalo buffalo.

- Bison intimidate bison from Buffalo.

Buffalo buffalo buffalo buffalo.

- Bison intimidated by bison intimidate bison.
- Bison from Buffalo intimidate bison.



Additional reasons

- Stories are comprehensible
 - Developers and customers understand them
 - People are better able to remember events if they are organized into stories[†]
- Stories are the right size for planning
- Stories support and encourage iterative development
 - Can easily start with epics and disaggregate closer to development time

Yet more reasons

- Stories support opportunistic development
 - We design solutions by moving opportunistically between top-down and bottom-up approaches[†]
- Stories support participatory design
 - Participatory design
 - The users of the system become part of the team designing the behavior of the system
 - Empirical design
 - Designers of the new system make decisions by studying prospective users in typical situations

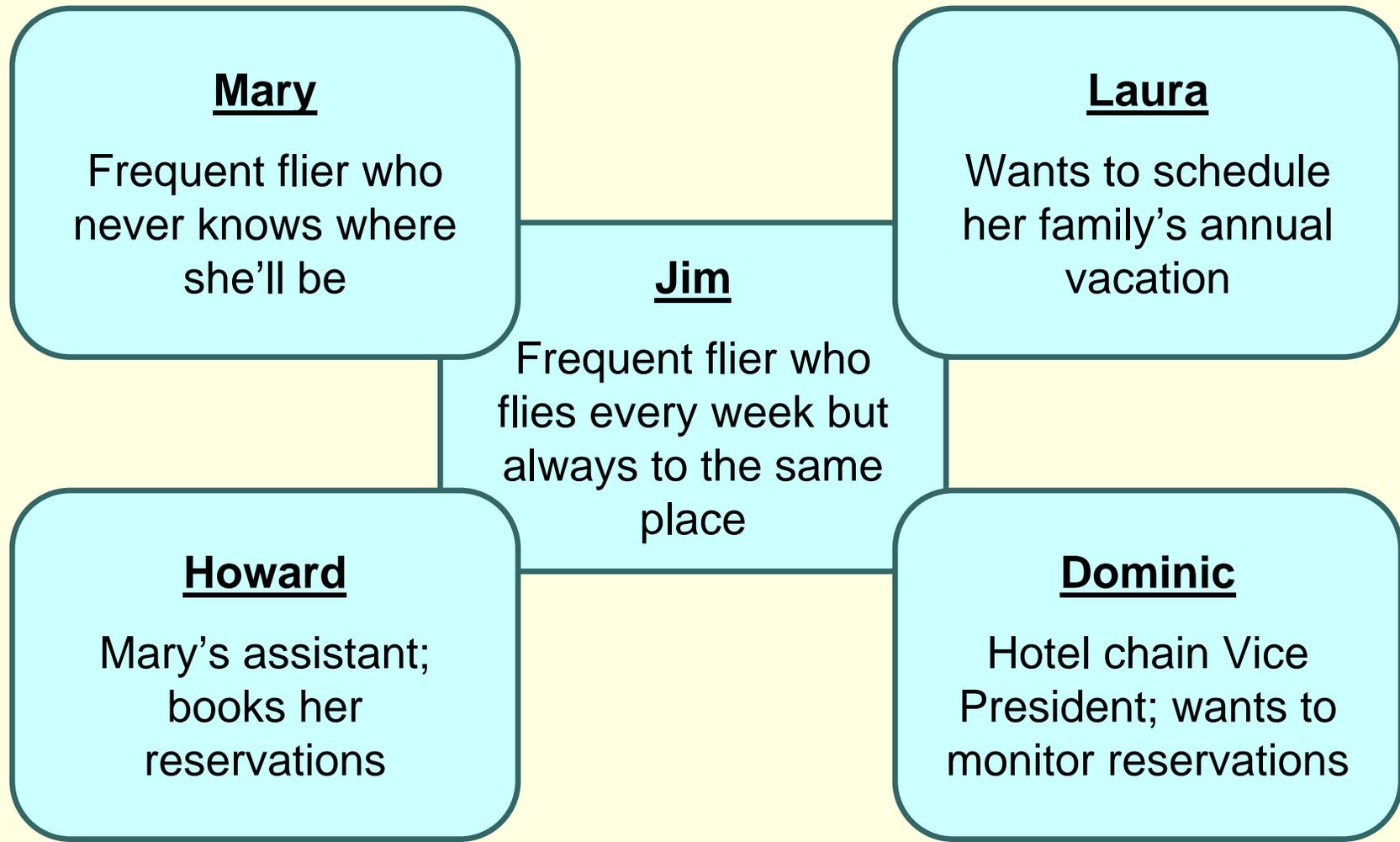
Today's agenda

- What are user stories?
- Why user stories?
- User role modeling
- INVEST in good stories
- Guidelines for writing good stories

“The User”

- Many projects mistakenly assume there's only one user:
 - “The user”
- Write all stories from one user's perspective
- Assume all users have the same goals
- Leads to missing stories

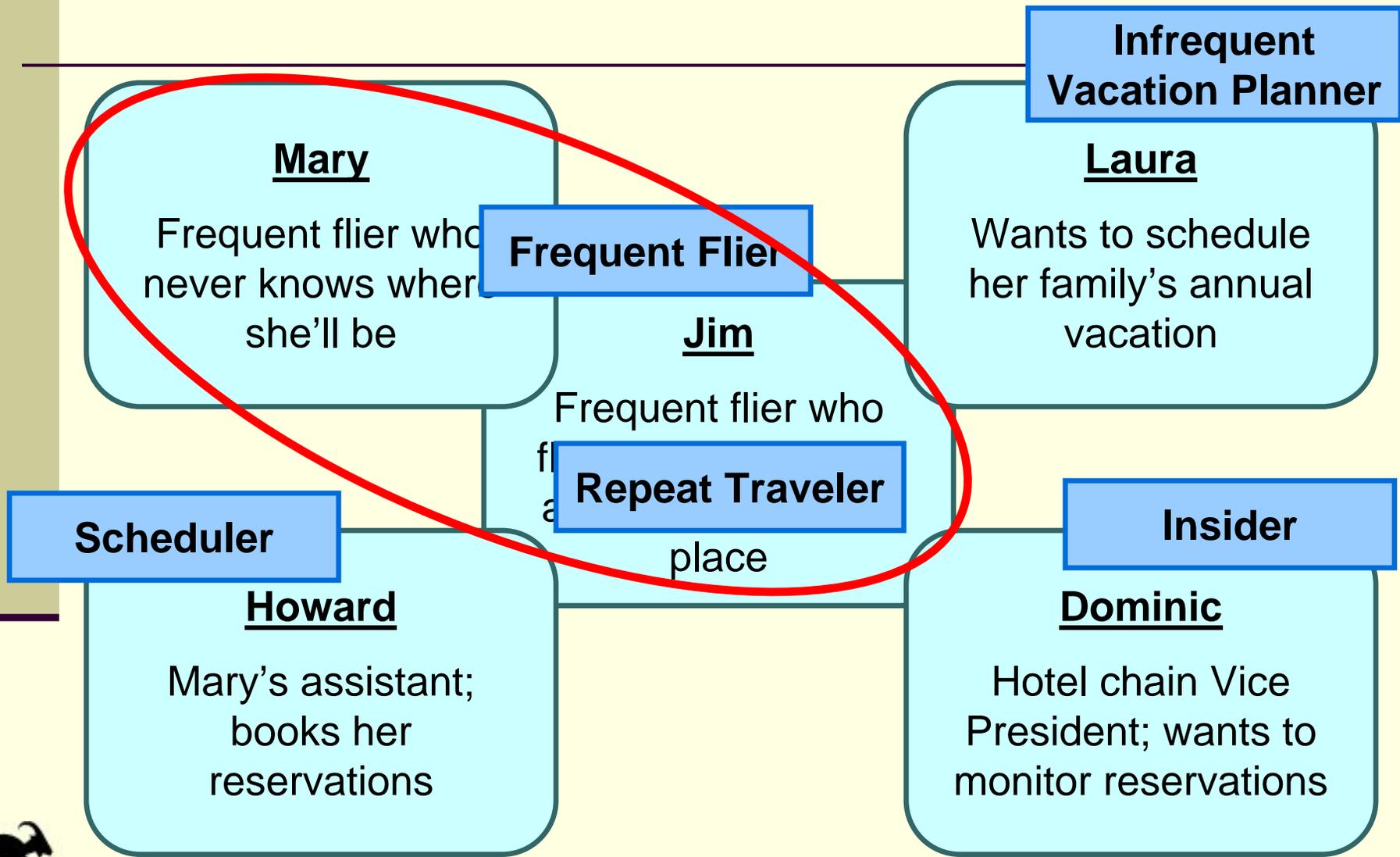
Travel Site—Who's the user?



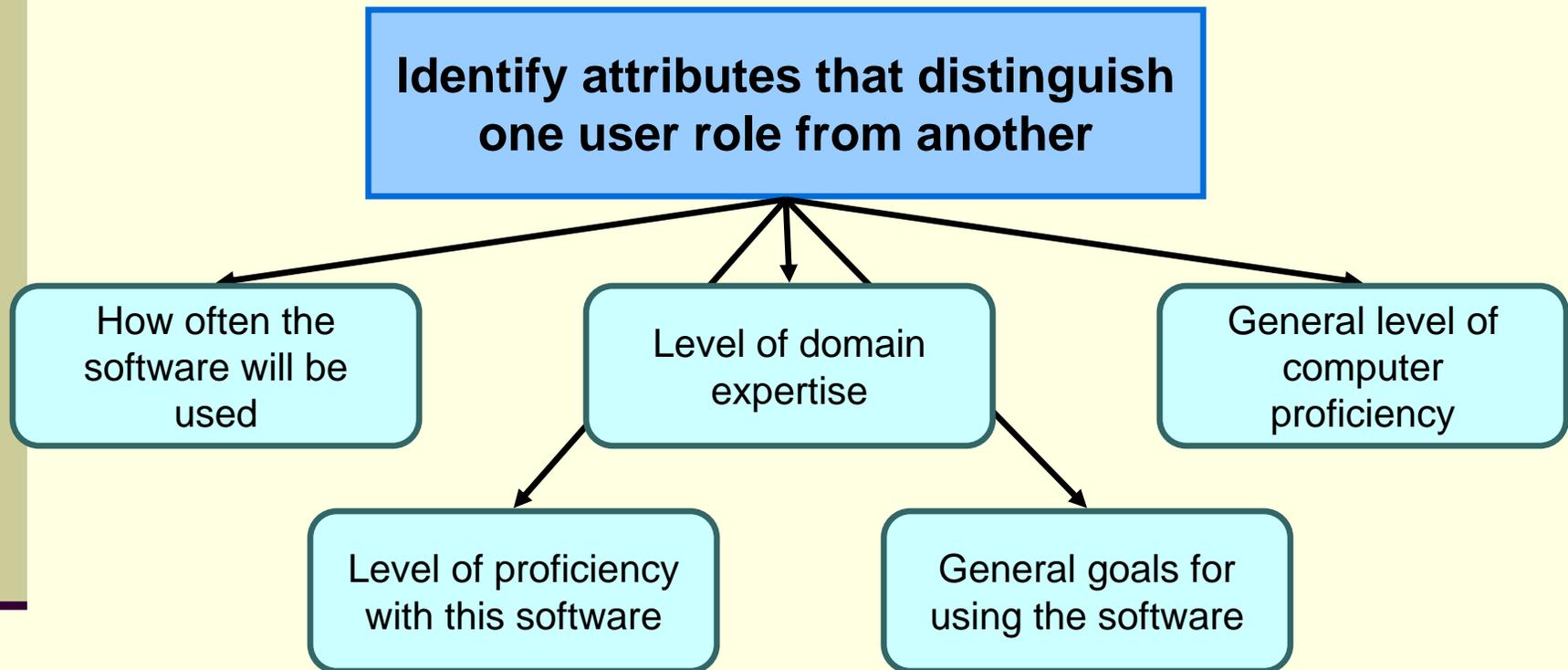
User roles

- Broaden the scope from looking at one user
- Allows users to vary by
 - What they use the software for
 - How they use the software
 - Background
 - Familiarity with the software / computers
- Used extensively in usage-centered design
- Definition
 - A user role is a collection of defining attributes that characterize a population of users and their intended interactions with the system.

Common attributes



User role modeling



Document the user role

User Role: Infrequent Vacation Planner

Not particularly computer-savvy but quite adept at using the web. Will use the software infrequently but intensely (perhaps 5 hours to research and plan a trip). Values richness of experience (lots of content) over speed. But, software must be easy to learn and also easily recalled months later.

Personas

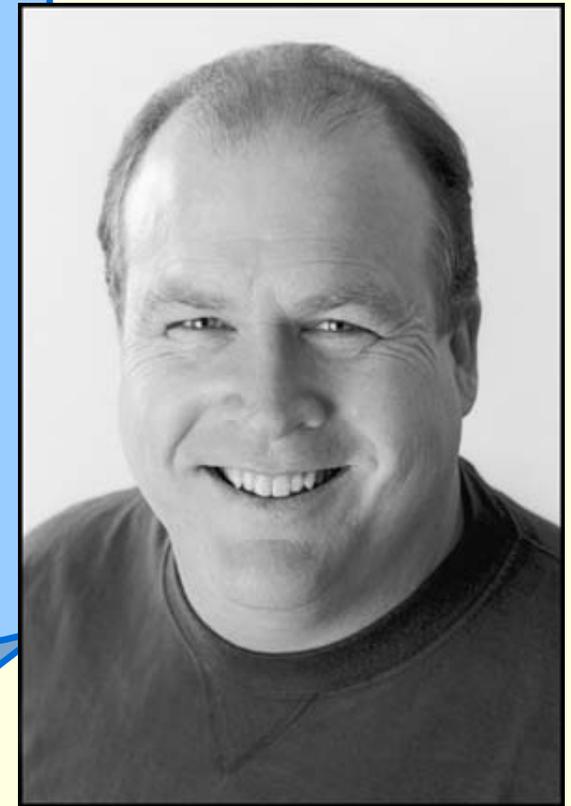
- A central element of Alan Cooper's interaction design
- A persona is an imaginary representation of a user role
- A natural extension to user roles
- Generally, avoid picking personas who are real users

Add details to each persona

- Likes, dislikes
- When, where, why
- Model and make of car
- Job
 - Not “is a florist” but “works as a florist at Lake Park Florist”)
- Goals
 - Not “planning a vacation but “planning the family vacation to Yellowstone”

A sample persona

Jim lives in four bedroom house in a nice suburb north of Chicago. However, he works as a vice president of marketing in Sacramento, California. Three weeks out of every four he flies from Chicago to Sacramento on Monday morning and then flies home on Friday. The company lets him work every fourth week out of his home. Jim schedules his own flights, usually a month or more in advance. He's partial to United Airlines but is always on the lookout for bargain fares so that the company will allow him to continue to live in Chicago. Jim quickly learns most software but becomes very impatient when he finds a bug or when a website is slow.



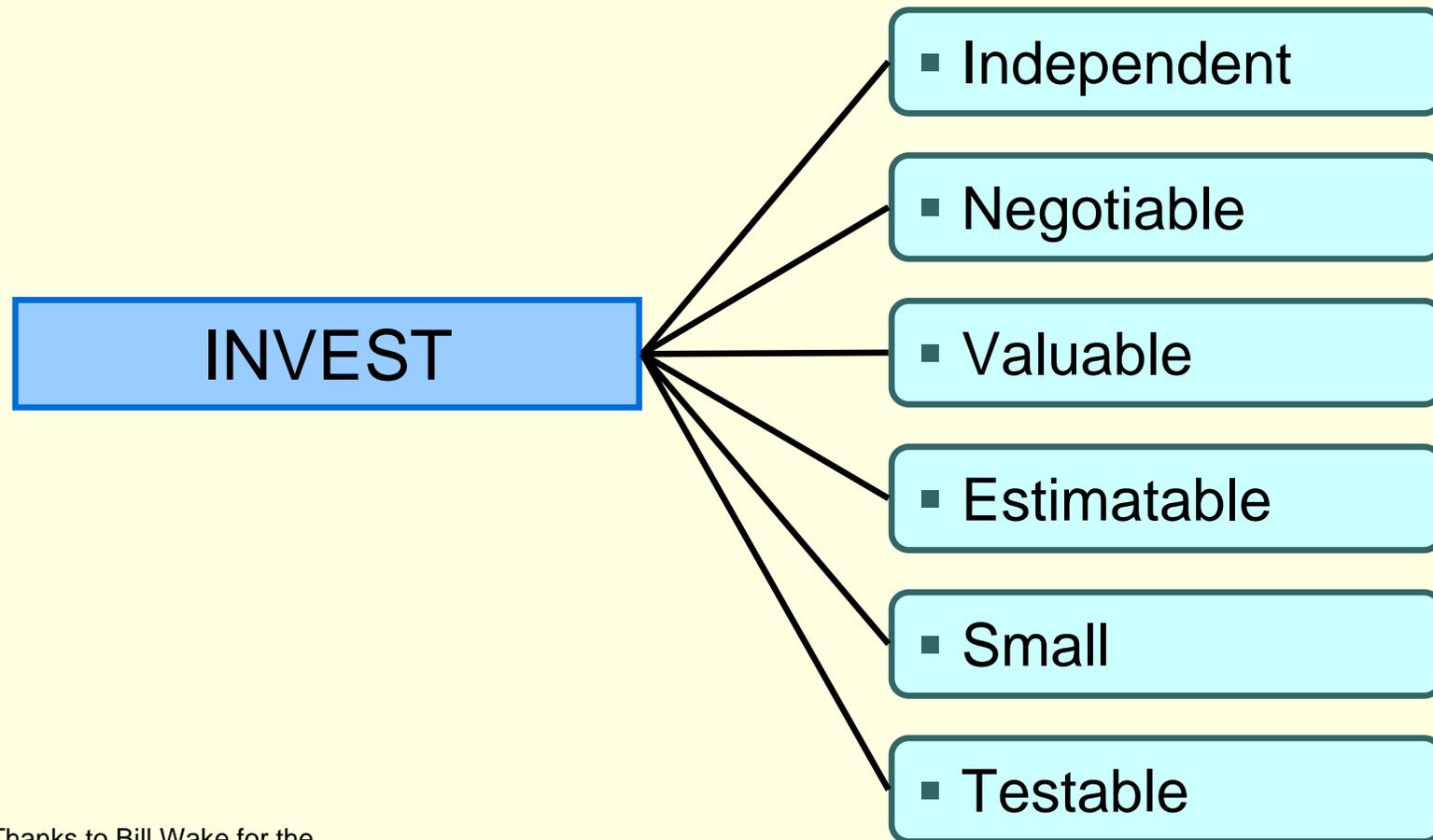
Why do user role modeling

- Start thinking of the software as solving the needs of real people
- Avoid saying “the user” and instead say
 - “A Frequent Flier...”
 - “A Repeat Traveler...”
 - “Jim...”

Today's agenda

- What are user stories?
- Why user stories?
- User role modeling
- INVEST** in good stories
- Guidelines for writing good stories

What makes a good story?



Thanks to Bill Wake for the acronym. See www.xp123.com.



Independent

- Avoid introducing dependencies
 - Leads to difficulty prioritizing and planning

A company can pay for a job posting with a Visa card.

?

A company can pay for a job posting with an AmEx card.

A company can pay for a job posting with a MasterCard.

?

- The first of these stories will take 3 days to develop
 - It doesn't matter which is first
- The others will take 1 day



Making stories independent

Combine the stories

- A customer can pay with a credit card.

Split across a different dimension

- A customer can pay with one type of credit card.
- A customer can pay with two other types of credit cards.

Write two estimates and move on

- 3 days if first; 1 otherwise

Negotiable

- Stories are not
 - Written contracts
 - Requirements the software must fulfill
- Do not need to include all details
- Too many details give the impressions of
 - false precision or completeness
 - that there's no need to talk further
- Need some flexibility so that we can adjust how much of the story gets implemented
 - If the card is contract then it needs to be estimated like a contract

Is this story negotiable?

A company can pay for a job posting with a credit card.

Note: Accept Visa, MasterCard, and American Express. Consider Discover. On purchases over \$100, ask for card ID number from back of card. The system can tell what type of card it is from the first two digits of the card number. The system can store a card number for future use. Collect the expiration month and date of the card.



How about this one?

A company can pay for a job posting with a credit card.

Note: Will we accept Discover cards?

Note for UI: Don't have a field for card type (it can be derived from first two digits on the card).



Valuable

- Stories must be valuable to either:

Users

- A user can search for a job by title and salary range.

Purchasers

- Throughout the project, the development team will produce documentation suitable for an ISO 9001 audit.
- The development team will produce the software in accordance with CMM level 3.
- All configuration information is read from a central location.

Stories valued by developers

- Should be rewritten to show the benefit

All connections to the database are through a connection pool.



Up to 50 users should be able to use the application with a five-user database license.

All error handling and logging is done through a set of common classes.



All errors are presented to the user and logged in a consistent manner.



Estimatable

- Because stories are used in planning
- A story may not be estimatable if:

Developers lack domain knowledge

- New users are given a diabetic screening.

Developers lack technical knowledge

- A user can select to see all text on the site in a larger font.

The story is too big

- A user can find a job.

Small

- Large stories (epics) are
 - hard to estimate
 - hard to plan
 - They don't fit well into single iterations
- Compound story
 - An epic that comprises multiple shorter stories
- Complex story
 - A story that is inherently large and cannot easily be disaggregated into constituent stories

Compound stories

- Often hide a great number of assumptions

A user can post her resume.

- A resume includes separate sections for education, prior jobs, salary history, publications, etc.
- Users can mark resumes as inactive
- Users can have multiple resumes
- Users can edit resumes
- Users can delete resumes

Splitting a compound story

Split along operational boundaries (CRUD)

- A user can create resumes, which include education, prior jobs, salary history, publications, presentations, community service, and an objective.
- A user can edit a resume.
- A user can delete a resume.
- A user can have multiple resumes.
- A user can activate and inactivate resumes.

Splitting a compound story, cont.

Split along data boundaries

- A user can add and edit educational information on a resume.
- A user can add and edit prior jobs on a resume.
- A user can add and edit salary history on a resume.
- A user can delete a resume.
- A user can have multiple resumes.
- A user can activate and inactivate resumes.

Testable

- Tests demonstrate that a story meets the customer's expectations
- Strive for 90+% automation

A user must find the software easy to use.



A novice user is able to complete common workflows without training.

A user must never have to wait long for a screen to appear.



New screens appear within 2 seconds in 95% of all cases.



Today's agenda

- What are user stories?
- Why user stories?
- User role modeling
- INVEST in good stories
- Guidelines for writing good stories

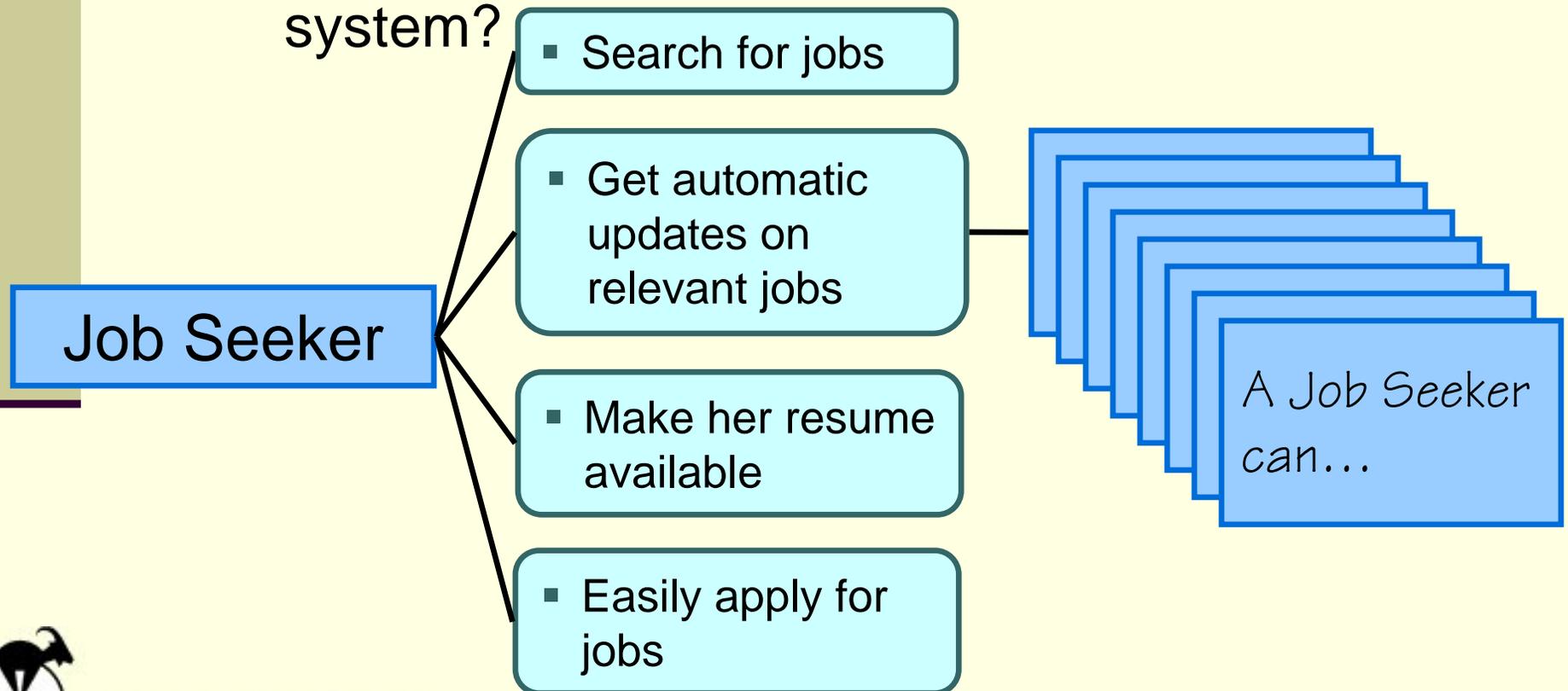
Additional guidelines for good stories

- Start with goals
- Slice the cake
- Write closed stories
- Put constraints on cards
- Size the story to the horizon
- Keep the UI out as long as possible
- Some things aren't stories
- Include user roles in the stories
- Write for one user

Don't forget the purpose

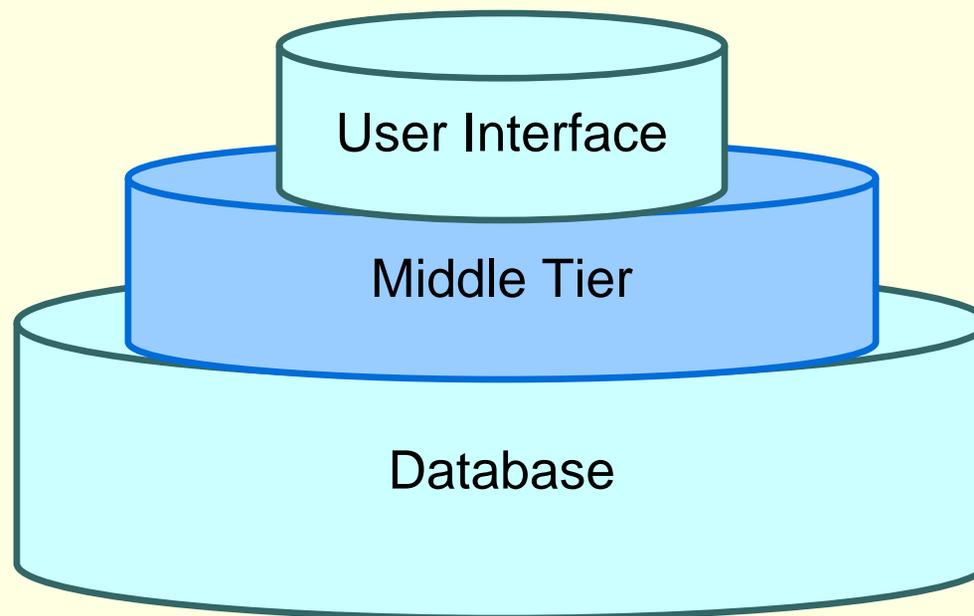
Start with goals

- For each role, ask
 - What are this user's goals in using the system?



Slice the cake

- Our first inclination is often to write stories that are purely from one layer
- We're better off taking a slice through the entire cake



An example

- These stories do not “slice the cake”:

A Job Seeker can post a resume.

- A Job Seeker can fill out a resume form.

- Information on a resume form is written to a database.

A better way

A Job Seeker can post a resume.

- A Job Seeker can submit a resume that includes only basic information such as name, address, and education history.
- A Job Seeker can submit a resume that includes all information an employer may want to see.

Why?

- Exercising each layer reduces architectural risk
- Easier to prioritize
 - Stories that don't slice the cake tend not to provide any business value
- Application could be released early with only a few slices done

Write closed stories

- A closed story is one that finishes with the achievement of a meaningful goal.
 - User feels she's accomplished something.

A user can manage the ads she's placed.

- This story is never done
- It's something the user does on an ongoing basis

Examples of closed stories

A user can manage the ads she's placed.

- A recruiter can review resumes from applicants to one of her ads.
- A recruiter can change the expiration date of an ad.
- A recruiter can delete an application that is not a good match for a job.

Put constraints on cards

- Write constraints on cards, just like any other stories
- Annotate with “constraint.”
- Put each into the earliest possible iteration
- Have tests to verify the constraint is met

The system must support peak usage of up to 50 concurrent users.

Constraint

More example constraints

- Do not make it hard to internationalize the software if needed later.
- The new system must use our existing order database.
- The software must run on all versions of Windows.
- The system will achieve uptime of 99.999%.
- The software will be easy to use.

Size the story to the horizon

- Focus attention where it's needed most
- If the story will be coded soon,
 - Write stories that can be estimated and used in planning
- If not,
 - Write an epic
- Strive for a system where developers *pull* stories through the system
 - Rather than where stories *push* developers to go faster

Keep the UI out as long as possible

- On a new project the UI doesn't exist, so leave it out of stories as long as possible
- Including UI detail in a story constrains the possible solutions
- Eventually, you'll have UI-specific stories:
 - “Add a page size button to the print dialog.”
 - “Take some fields on the search screen and hide them behind a ‘more...’ button.”

Too much UI detail

Print dialog allows the user to edit the printer list. The user can add or remove printers from the printer list. The user can add printers either by auto-search or manually specifying the printer DNS name or IP address. An advanced search option also allows the user to restrict his search within specified IP addresses and subnet range.

Some things aren't stories

- If you have a requirement that doesn't fit as a story, write something else
 - A use-case
 - User interface guidelines
 - A list of business rules
 - Interface with another system
- Whatever you write, keep it lightweight

Include user roles in the stories

- Sometimes all users want to act in a specific story but often it's a type of user
- Help everyone by putting that user in mind when looking at the story card:
 - A Job Seeker can post a resume.
 - A Recruiter can read submitted resumes.
- A template I really like to start with:
 - “As a <role> I want to <story> so that <benefit>.”

Write for one user

- Usually it doesn't matter:

Recruiters can search for good candidates.

- But often enough it causes confusion:

Job Seekers can post resumes.

- Can one job seeker post multiple resumes?



Single-user stories remove ambiguity

- Written for one user, it's clear that each user can post multiple resumes

Job Seekers can post resumes.



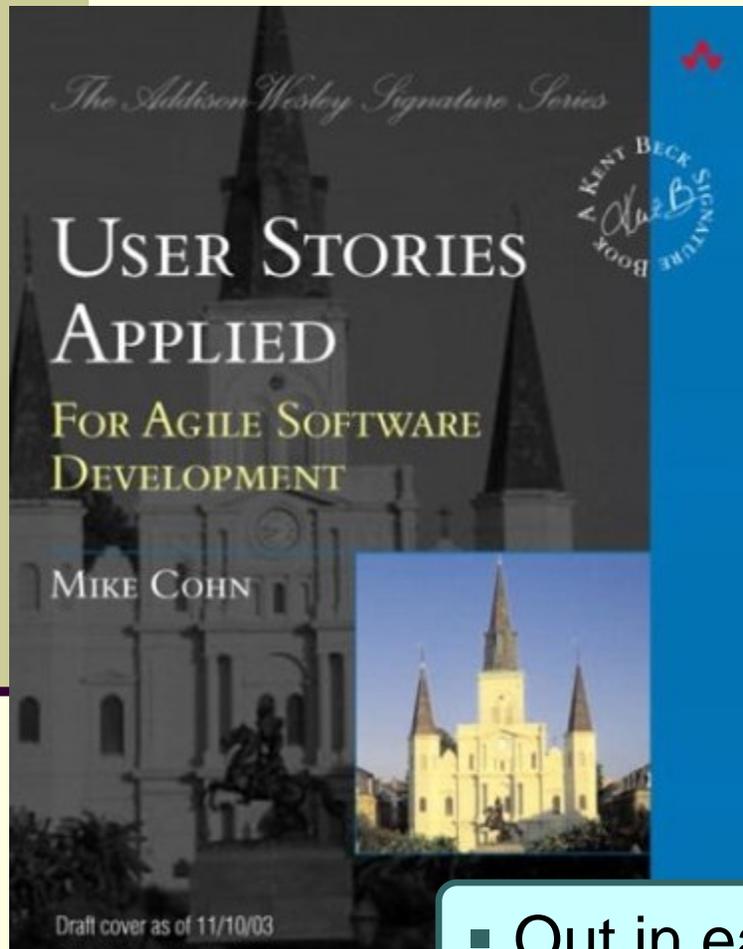
A Job Seeker can post resumes.

Most importantly...

Don't forget the
purpose

- The story text we write on cards is less important than the conversations we have.
- “Stories represent requirements, they do not document them.”[†]

For more on user stories



- Software Development West
 - March 15:
 - Half day tutorial on user stories
 - March 17:
 - 90-minute class on agile estimating and planning

■ Out in early March



Where to go next?



User Stories

- groups.yahoo.com/userstories
- www.userstories.com

Agile in General

- www.agilealliance.com

Scrum

- www.mountangoatsoftware.com/scrum

My contact information

Email

- mike@userstories.com
- mike.cohn@computer.org

- www.userstories.com
- www.mountangoatsoftware.com

Websites