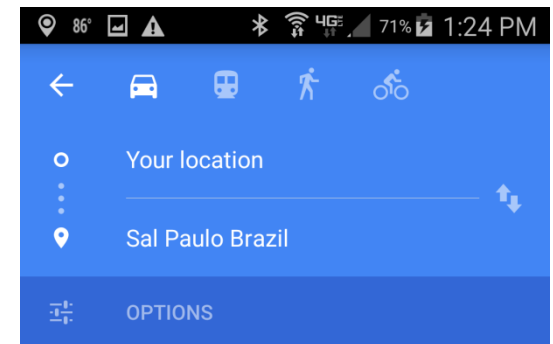




Wanted: A GPS to Track My Agile Project

{like, where is it, really}?



No route found

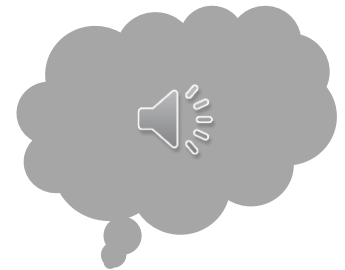
Joe Schofield

joescho@joejr.com

How is my house / apartment?

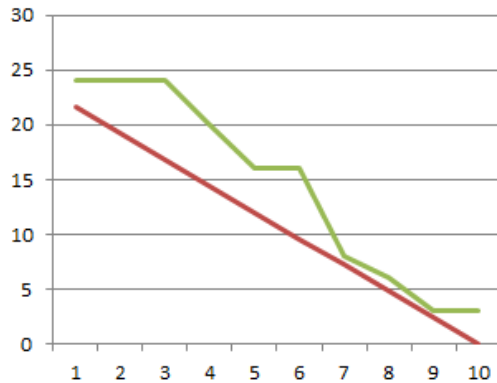


- You can't always believe what you think you see
- The same is true with software productivity
- Agile software development status and productivity are subject to the same scrutiny
- Technological change makes analysis and understanding even more difficult



What “story” does the burndown chart tell?

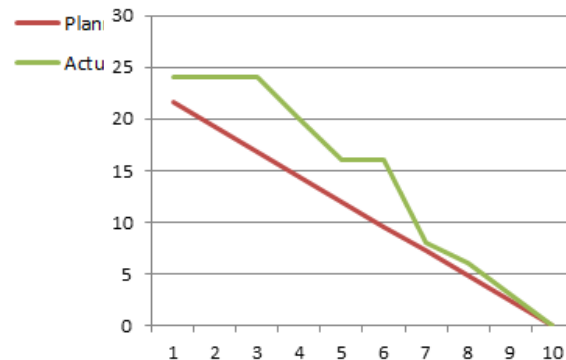
10-Day Burndown with Story Points



The team may have:

(Using velocity and story points as intended)
Under-estimated (the story points) or under-performed or both

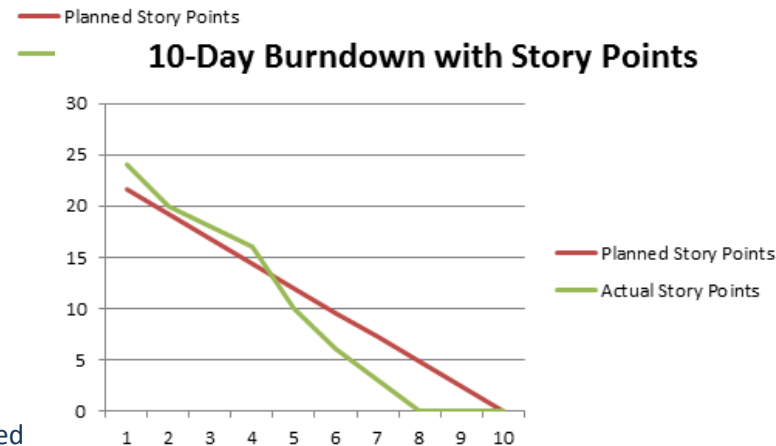
10-Day Burndown with Story Points



The team may have:

- A. (Using velocity and story points as intended) performed exactly as estimated
- B. (Inflating the story points) Completed the same or less work than on previous sprints while giving the appearance of completing the same amount of work

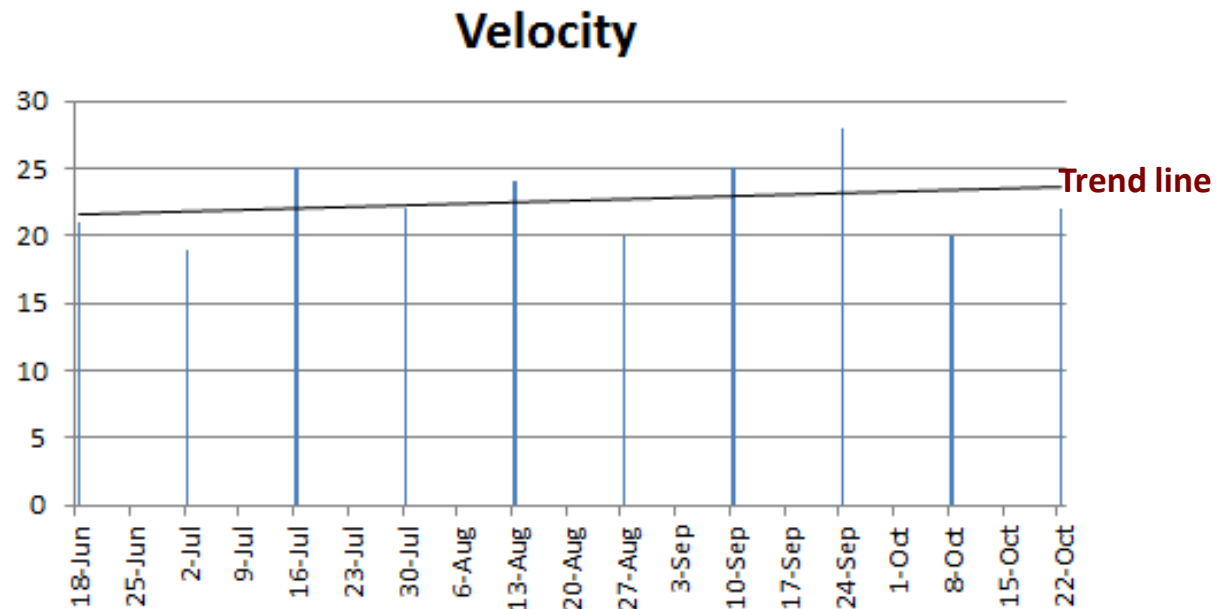
10-Day Burndown with Story Points



The team may have:

- A. (Using velocity and story points as intended) performed spectacularly beating the estimates
- B. (Inflating the story points) Completed the same or less work than on previous sprints while giving the appearance of over-achieving by finishing the sprint two days early

What “story” does the velocity chart tell?



The team may:

- A. (Using velocity and story points as intended) experiencing a slight increase in velocity over time
- B. (Inflating the story points) Completed the same or less work than on previous sprints while giving the appearance of completing an increasing amount of work

Questions to be Answered . . .

- What exactly is “agile” development? In a recent Rally webinar poll 71% . . .
- What elements of culture may be impeding agile adoption?
- What confidence should I have in agile estimates?
- Can I get true project status without using a tracking device or drone?
- Can I demonstrate team productivity improvements?
- Is my process trending better or worse?
- Can function points improve objectivity in reporting progress?
- Am I alone in overcoming these challenges or are there best agile practices?

Abstract

Abstract: Culture, estimating, funding, tracking, sizing, productivity, quality, and measurement of each of these are challenges that can be mitigated by informed organizations, but left to chance these will undermine an organization's transition to agile adoption. Answers to the following questions differentiate the informed from those in need of a GPS:

- What exactly is agile? You will be surprised.
- What cultural impediments are slowing my success?
- Why the tension with the PMO (Project Management Office)? Got none (or a good one), you're lucky.
- How confident should I be with agile estimates?
- Can I know the project status without getting a drone to stalk track the team's progress?
- Can I prove that my team is more productive than in the past?
- Is my product better? Is my process improving?
- How can I use my function point expertise to help with agile objectivity?
- Are my challenges new or can I find answers among the fossils?
- What's the best of the best agile practices?

Finally, what happens when we combine the world's most used software development approach with the world's most used functional measurement analysis with an audience that represents the world's largest software functional measurement certified population? Let's find out!

Questions to be Answered . . .

What exactly is “agile” development?

- **Scrum – 56%**
- **Extreme programming (XP) (with Scrum) – 10%**
- **Scrumban – 6%**
- **Kanban (development) – 5%**
- **Lean software development – 2%**
- **All others ~1% or less**
 - **Adaptive software development (ASD)**
 - **Agile modeling**
 - **Agile Unified Process (AUP)**
 - **Business analyst designer method (BADM)**
 - **Crystal Clear Methods**
 - **Disciplined Agile Delivery (DAD)**
 - **Dynamic systems development method (DSDM)**
 - **Feature-driven development (FDD)**

9th Annual State of Agile Survey, VersionOne

Answers to “Questions to be Answered”

What elements of culture may be impeding agile adoption?

- **Traditional project planning and budgeting (“hard” milestones, delivery dates, and fixed \$)**
- **Traditional roles (analyst, developer, architect, tester vs. team member)**
- **Traditional organization structures: NEED: a few good SILO BUSTERS**
- **HR – hiring for roles and typecasting**
- **Adopting and swapping tools without understanding intended practices**

Answers to “Questions to be Answered”

How much confidence should I have in agile estimates?

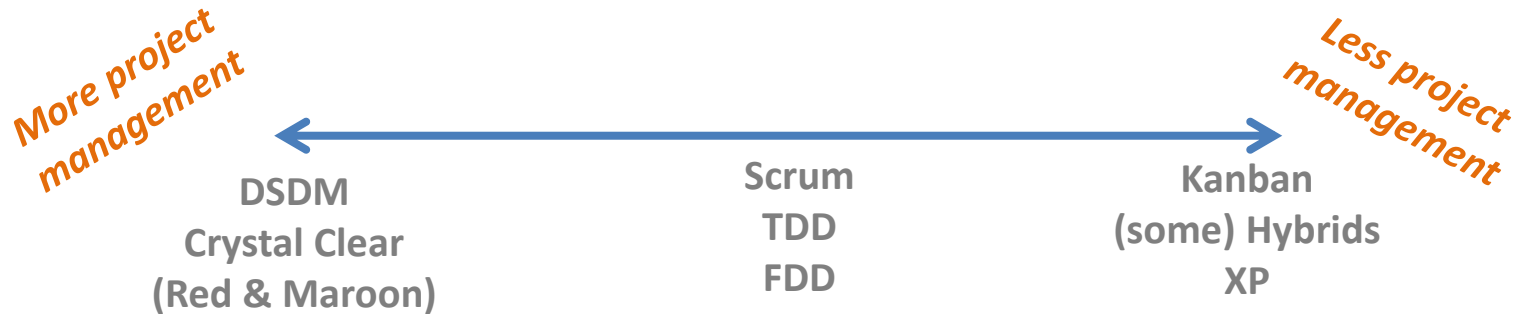
- Questions were raised around this question already (see slides 3 & 4)
- Use velocity for items sized in the product backlog (see slide 11 (soon))
- Associate story points to time estimates (although you don't know the time associated with a story until it's decomposed into “tasks”)
- Associate a labor time metric to story points (agile purists may not like that since they intend no such relationship)



Dan Chuparkoff,
4/21/2014

Answers to “Questions to be Answered”

- Can I get true project status without using a tracking device or drone?
(YES, NO, MAYBE, it depends)



- Can I demonstrate team productivity improvements?
- Is my process trending better or worse?
- Can function points improve objectivity in reporting progress?
Answer: “yes” for stories that are decomposed to a transactional level
(example on next slide)

Potential Size Measurement Approaches with Scrum



Portfolio /
Enterprise Backlog



Use:

Rock, scissors,
paper

*Approximate based on
Sprint Backlog*

**Estimate based on
Sprint Backlog*

**Measure delivered
Function Points and velocity*

Compare to:

- \$\$\$ spent
- Time / schedule
- Latent defect rate
- Defect removal efficiency

An example:

Assume six stories in the Sprint Backlog are sized at 76 (13 + 21 + 8 + 21 + 5 + 8)

Let's also assume these story point are 59 function points (15, 15, 10, 7, 7, 5)

- *Estimate* the Release Backlog using function point analysis (looks like about 10 FPs per story or about 90 in this case)
- *Approximate* the product backlog using the stories that are "sprint ready" at the top of the Product Backlog and the team's best understanding of the remaining stories and epics in the Product Backlog (STOP HERE!) (about 1/3 of the product backlog is approximated @ 40 function points)
- Use "rock, scissors, paper" as a poor attempt to measure something that should not be measured!

*assumes stories are at a transactional level

November 18, 2015

Joe Schofield – ISMA 11

Answers to “Questions to be Answered”

Am I alone with these challenges or are there best agile practices?

- Visualize the process with physical movement – beware of tool substitution corrupting visibility and “high touch / low tech” thinking
- Use story points consistently throughout project – beware of inflation⁰
- Trust recent velocity – beware of “optimism”¹ and pessimism
- Churnover (changing team members) still kills²
- Stay pure – hybridaphobia (fear of hybrids) can be good
- Let folks perform their roles; don’t need extra Product Owners or Scrum Masters
- A recent CAST study found that organizations that did architectural planning *before* starting their agile projects were more successful than those practicing agile or waterfall methodologies alone; is this “watergile”?⁴
- Ten more reasons in “*Keep the Baby*”⁵

⁰*Inflate Gate: Mastering Overestimation for Agile Software Projects*; Computer Aid’s Accelerating IT Success, featured article; 8/3/2015; Schofield

¹Underestimation in the “When It Gets Worse Before it Gets Better” Phenomenon in Process Improvement; Advanced Concurrent Engineering, 2011, Part 1, 3-10, DOI: 10.1007/978-0-85729-799-0_1; Ricardo Valerdi and Braulio Fernandes

²Tuckman’s Team Model; https://en.wikipedia.org/wiki/Tuckman's_stages_of_group_development; retrieved 10/24/2015

⁴*Keep the Baby*; Schofield; MetricViews; January, 2015

⁵*The CRASH Report 2014 – 2015, Executive Summary*; CAST

Answers to “Questions to be Answered”

9th Annual State of Agile Survey, VersionOne

Why use agile?

Reported reasons for adopting agile (VersionOne):

- 59% - Faster product delivery
- 56% - Manage changing priorities
- 53% - Increase productivity
- 46% - Enhance product quality
- 40% - Improve IT to business alignment
- 40% - Improve project visibility
- 38% - Reduce project risk
- 26% - Improve team morale
- 25% - Improve engineering discipline
- 23% - Reduce project costs
- 22% - Improve software support
- 20% - Manage distributed teams

Reported success with Agile (VersionOne):

- 58% - Delivery timeliness
- 48% - Improve product quality
- 44% - C-sat
- 44% - Business Value
- 30% - Project visibility
- 29% - Productivity
- 25% - Predictability
- 23% - Process Improvement

Reported success with Agile (Rally):

- 29% - Better cost,
- 91% - Better schedule,
- 97% - Better productivity,
- 50% - Better quality,
- 400% - Better satisfaction, and
- 470% - Better ROI than the least effective (traditional) ones?

Wrong Motives, Bad Practices

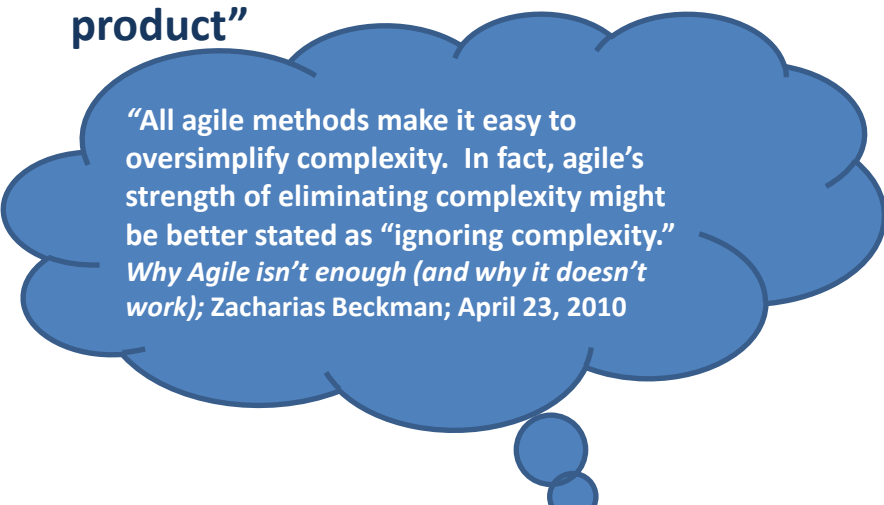
(we're still novices at applying agile concepts)

(don't) Use "agile" because:

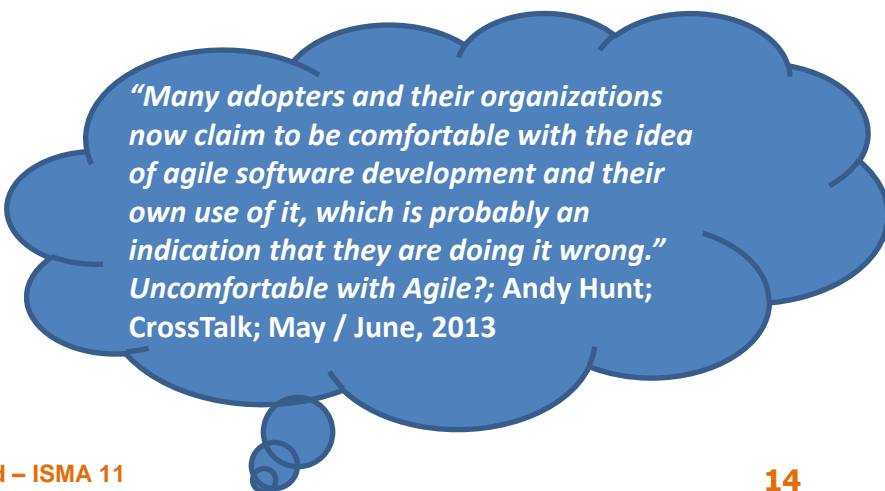
- Everybody else is doing it
- Somebody else is doing it (unless you want to be them!) Example: "So and so" is doing everything with agile . . .
- "Waterfall" didn't work

(bad) Sprint practices:

- Postpone sprint demonstrations until sprint stories are ready for prime time (let sprints run until the work is done)
- Change sprint scope after it has started (adding stories to the sprint)
- Stack *test* sprints after *development* sprints – ignore the "potentially shippable product"



"All agile methods make it easy to oversimplify complexity. In fact, agile's strength of eliminating complexity might be better stated as "ignoring complexity."
Why Agile isn't enough (and why it doesn't work); Zacharias Beckman; April 23, 2010



"Many adopters and their organizations now claim to be comfortable with the idea of agile software development and their own use of it, which is probably an indication that they are doing it wrong."
Uncomfortable with Agile?; Andy Hunt; CrossTalk; May / June, 2013

Relevant Measurement Challenges with “Agile”

Estimation spans from dogs, to T-shirt sizes, to simple 1 to 10 scales, to Fibonacci scales – sounds good.

Story points provide ~~limited~~ no value beyond the team (as intended.)
Where is *organizational* learning – the 5th Discipline?

Other sources of estimation error during sprint planning⁰:

- Inflate story points to give an illusion of increased velocity
- Under reporting of productive time (%)
- Doubling “dipping” on what’s included in productive time (e-mails, phone calls, meetings) {count meetings a non-productive time, but then including them and their time in story tasks}
- Inflating the number of required tasks for story completion – “overloading” tasks with refactoring
- Inflating tasks time for story completion

⁰*Inflate Gate: Mastering Overestimation for Agile Software Projects*; Computer Aid’s Accelerating IT Success, featured article; 8/3/2015; Schofield

Story Points, Use Case Points, Function Points—What's the Point?

Characteristic	Function Points	Use Case Points	Story Points
Useful at the project level for estimating or planning	With historical FP data	With historical UCP data	With historical SP data
ISO / Standards based	ISO 20926	no	no
Captures customer view	Expected	Expected	Definitely
Useful for benchmarking outside the company	Could be	Could be	Less so
Easy to calculate	Less so	More so	Yes
Easy to validate for repeatability / consistency	More so	More so	Less so
Objectivity	More so	More so	Less so (team / team member variability)
Technologically independent	Yes	Yes	Maybe
Functional measurement to customer	Yes	Yes	Not exclusively (may include refactoring, design, and other work)

Deeper Reading *Function Points, Use Case Points, Story Points: Observations from a Case Study*; CrossTalk; May / June, 2013

An Industry Summary of Story Points as a Productivity Benchmark (across projects and across organizations!)

**This slide intentionally
left blank!**

Questions to be Answered . . .

- What exactly is “agile” development?
- What elements of culture may be impeding agile adoption?
- What confidence should I ascribe to agile estimates?
- Can I get true project status without using a tracking device or drone?
- Can I demonstrate team productivity improvements?
- Is my process trending better or worse?
- Can function points improve objectivity in reporting progress?
- Am I alone in overcoming these challenges or are there best agile practices?

So if you buy a GPS or drone to track your agile project . . .

- Once you've determined that it really is an agile project . . .
- You may be able to locate it, but you may not be able to determine its status
- You should be suspicious about any estimates until the team has earned the trust under which it operates
- *Productivity* will not reliably be comparable to historic projects, and possibly not to other active agile projects
- I hope your project is trending better; I'm not sure how you will know other than tracking value based on delivered product with each sprint
- Function points can help you determine size using an ISO standard
- You are not alone and “yes” there are best practices. Use them!

Further Reading

- *Inflate Gate: Mastering Overestimation for Agile Software Projects*; Computer Aid's Accelerating IT Success, featured article; 8/3/2015; Schofield
- *Keep the Baby*; Schofield; MetricViews; January, 2015
- *Keynote ISMA Cinco!*; Ricardo Valerdi
- Underestimation in the “When It Gets Worse Before it Gets Better” Phenomenon in Process Improvement; Advanced Concurrent Engineering, 2011, Part 1, 3-10, DOI: 10.1007/978-0-85729-799-0_1
- Ricardo Valerdi and Braulio Fernandes
- Decision-making during gambling: an integration of cognitive and psychobiological approaches; Luke Clark
- Measurements, Biases, Judgments: Understanding Variations for Reliable Estimates; Keynote Address - Sao Paulo, Brazil; Brazilian Software Measurement & Analysis Conference; November 11, 2011; Joe Schofield
- *Why We Should Measure Performance*; Oak Ridge Associated Universities, 2005
- *What Do We Measure and Why? Questions About The Uses of Measurement*; Journal for Strategic Performance Measurement; June 1999; Wheatley and Kellner-Rogers
- *How & Why We Measure Performance*; Yale New-Haven Hospital; <http://www.ynhh.org/patient-information/performance-measures.aspx>
- *How to Measure People's Performance*; Stacey Barr; staceybarr.com/questions/howtomeasurepeople/
- *Measuring People and Performance: Closing the Gaps*; Quality Texas; BS Morgan; 1999
- *Methods: Does Measuring People Change Them?*; The Psychologist, Vol. 24, Part 4; David P. French and Stephen Sutton
- MIT Sloan Sports Analytics Conference;
http://en.wikipedia.org/wiki/MIT_Sloan_Sports_Analytics_Conference;
- How the right measures help teams excel; C Meyer; Harvard Business Review ; 1994

About the presenter . . .



Joe Schofield is a past President of the International Function Point Users Group. He retired from Sandia National Laboratories as a Distinguished Member of the Technical Staff after a 31-year career. During twelve of those years he served as the SEPG Chair for an organization of about 400 personnel which was awarded a SW-CMM® Level 3 in 2005. He continued as the migration lead to CMMI® Level 4 until his departure. He continues that commitment today by helping his clients excel in agile transition, software and business improvement, and measurement.

Joe has facilitated over 100 teams in the areas of software specification, team building and organizational planning by using lean six sigma and business process reengineering. Joe has taught graduate courses since 1990. He was a licensed girl's mid-school basketball coach for 21 seasons--the last five undefeated, over a span of 50 games.

<http://joejr.com/presentd.htm> (~55)

<http://joejr.com/publishd.htm> (~36)

He has over 80 published books, papers, conference presentations and keynotes—including contributions to the books *The IFPUG Guide to IT and Software Measurement* (2012), *IT Measurement*, *Certified Function Point Specialist Exam Guide*, and *The Economics of Software Quality*. He is a CMMI Institute certified Instructor for the Introduction to the CMMI®, Certified Software Quality Analyst, Certified Function Point Specialist, a Certified Software Measurement Specialist, and most recently a Certified Scrum Master and Certified Agile Expert.

Joe is a frequent presenter at software measurement events—digitally and in person. He completed his Master's degree in MIS at the University of Arizona in 1980. By "others" he is known as a husband, father, son, and grandfather.