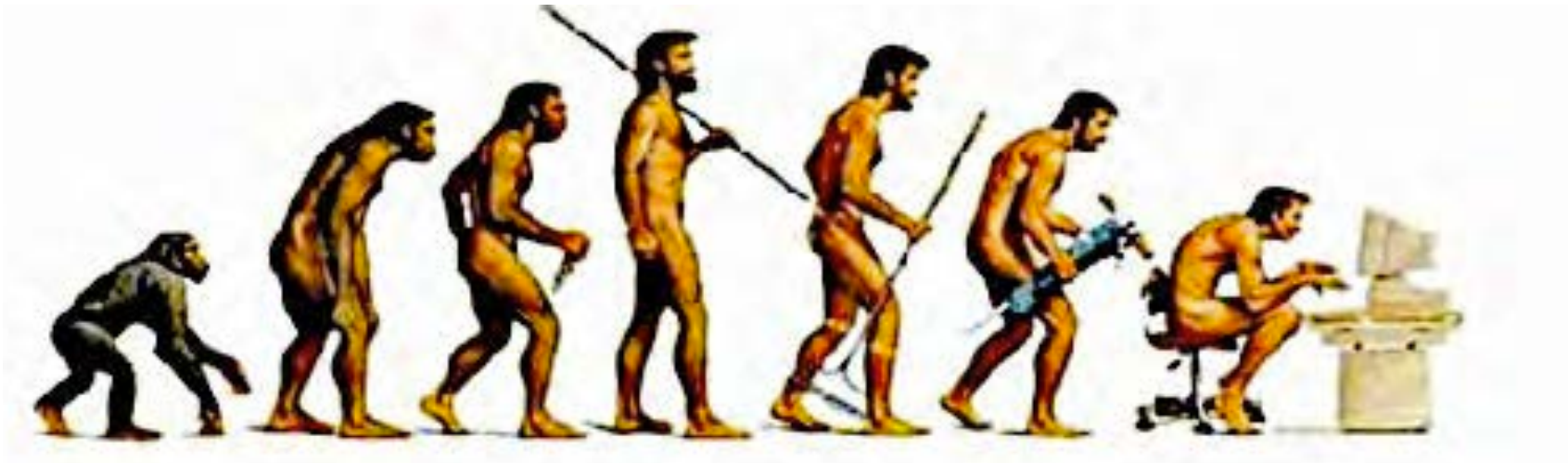
Web Services that Foster Innovation in Buildings Energy Analysis Tools

Evan Mills and Paul Mathew
Lawrence Berkeley National Laboratory

Presented at the 2012 ACEEE Summer Study on Energy Efficiency in Buildings, Asilomar, CA August 13-17

Thanks to the Home Energy Saver and EnergyIQ core teams:
Norm Bourassa, Leo Rainer, Rich Brown, Greg Homan, Danny Parker, and Andrea Mercado

Evolution of Energy Software Tools

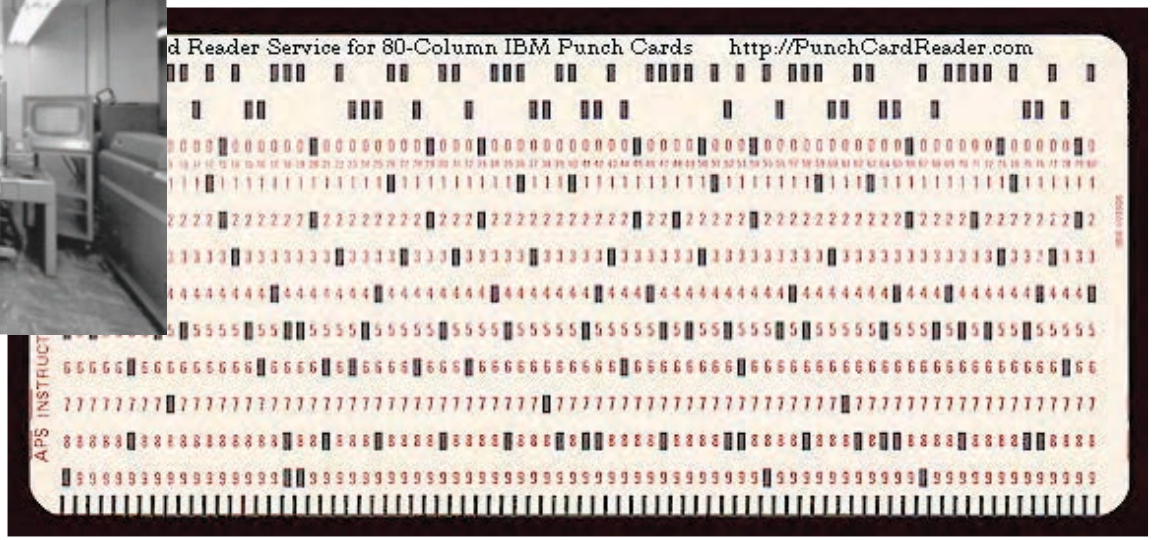


Hand Calculations

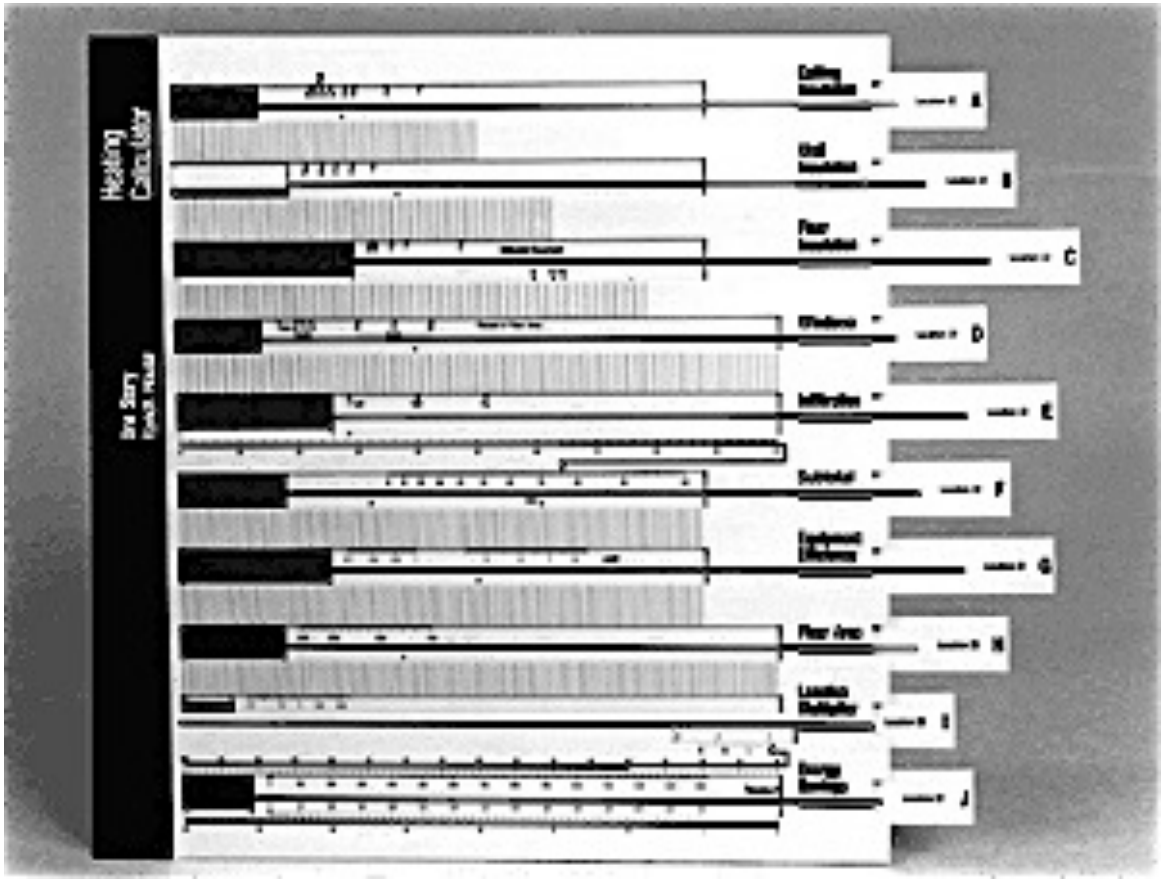
UAΔT + infiltration + solar

Mainframes

- Room-sized computers; painful input; no GUI; massive output
  - very narrow user base
  - runtime measured in <u>hours</u>



Lookups & Nomographs



Desktop Machines

- Disk-based
  - hundreds or thousands of users
  - runtime measured in minutes

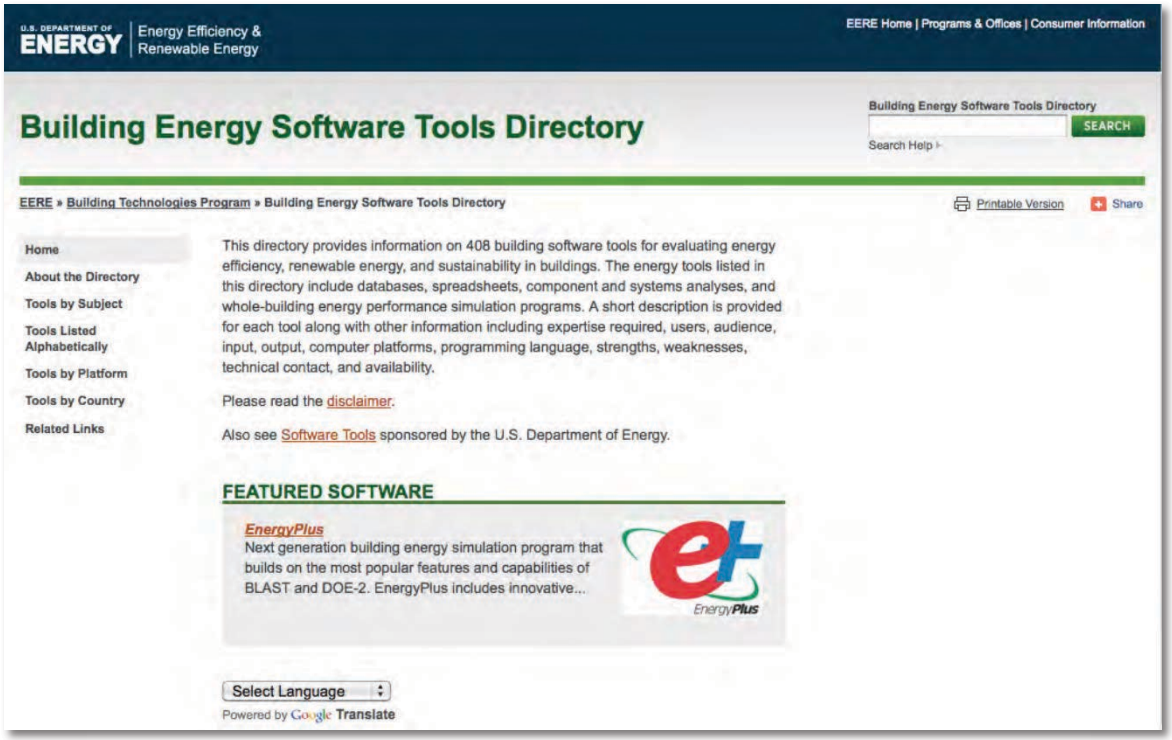
195kb capacity

Web-based

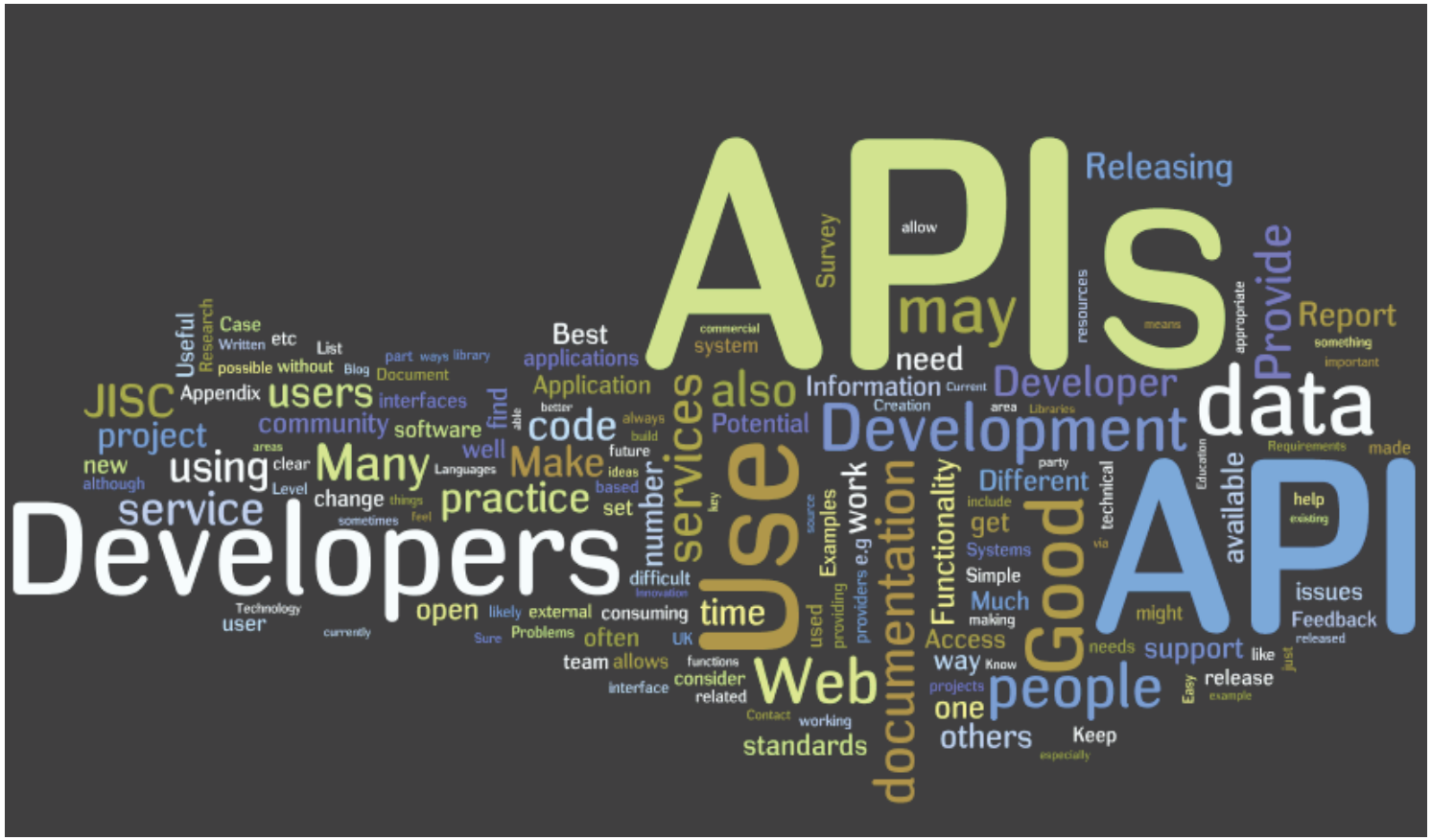
- Cloud-hosted applications with friendly GUIs
  - millions of users (including DIY)
  - runtime measured in seconds

At Least 400 Tools Now Exist

- Vibrant differentiation, but also a lot of redundancy in effort/cost (and inconsistency in results)

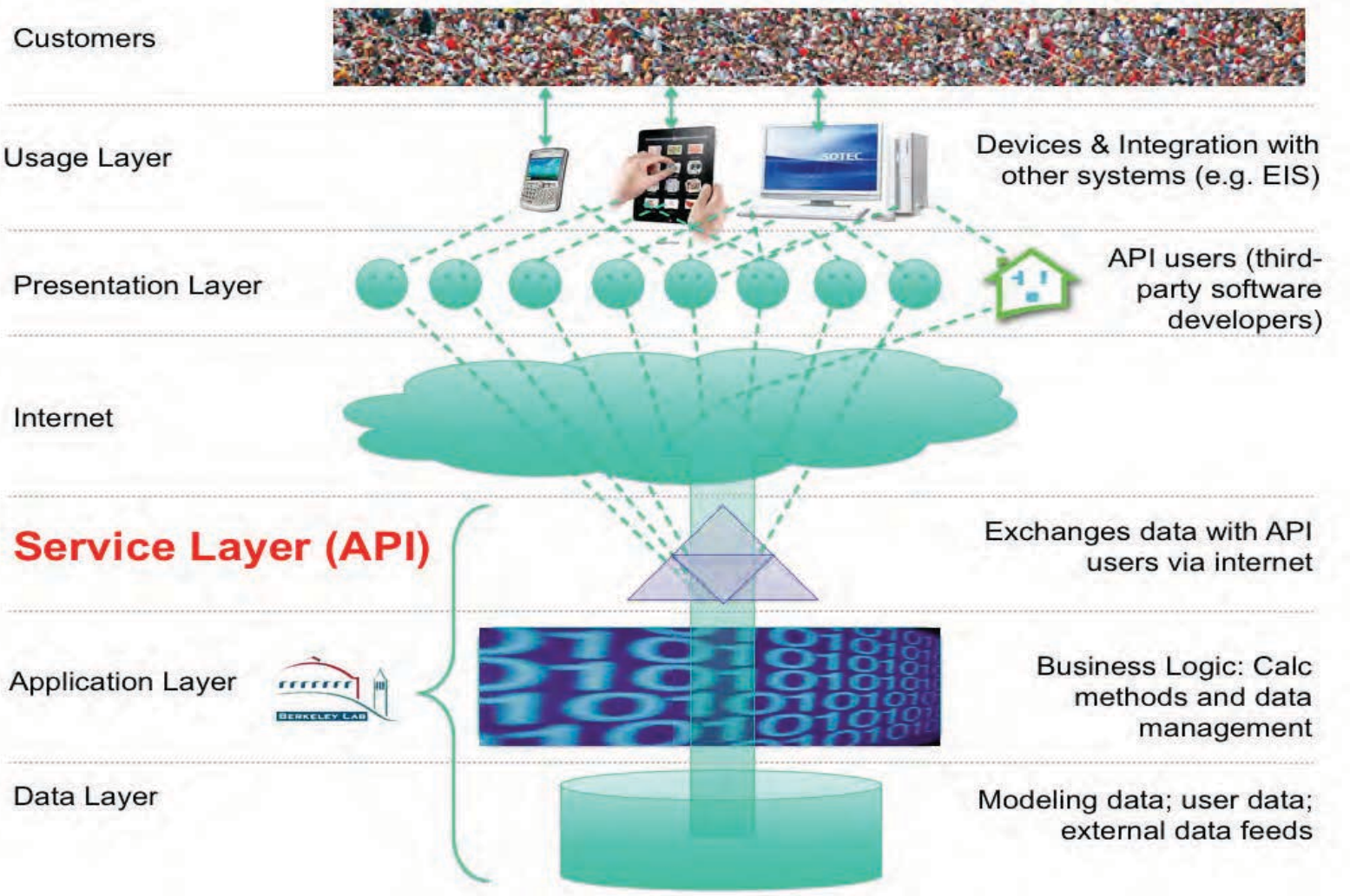


Software as a Service → APIs



What is an API, anyway?

APIs are protocols through which the a host computer and a client computer communicate and exchange data, leaving client free to decide how to gather input data from its customers and how to present them with results



APIs

Pros

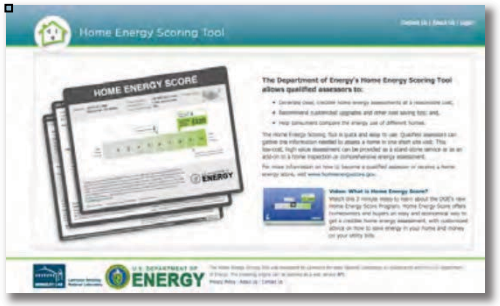
- Speeds and simplifies syndication of models and databases
- Radically lowers the cost of entry for private software developers
- Developers can focus more on front-end
- Enables more rapid innovation and differentiation of tools
- Facilitates more internal consistency in methodology and data across proliferation of tools
- Ameliorates stereotypical separation between “public” and “private” tools

Cons

- Requires web-infrastructure
- Derivative tools all depend on single API provider
- Initial development is slower; user support
- Developers need special skills and to be able to understand and adapt to outside service and support paradigm
- Requires very explicit documentation for third-party developers

LBNL APIs and Early Implementations

- **Home Energy Saver (Consumer & Pro versions)**
  - Whole-building simulation, including hourly HVAC (DOE2.1E)
  - Models/algorithms for all other end uses
  - Flexibility to model utilization and other behavioral variables
  - Defaults for every input
  - Whole building and end-use results: Energy / Costs / Emissions
  - Upgrade recommendations



- **Home Energy Scoring Tool**
  - HES Operational approach => Asset rating approach
  - Answers required (no user defaults)
  - Creates PDF label and upgrades list

- **EnergyIQ**
  - Non-residential energy benchmarking
  - 62 building types
  - 85 features/characteristics
  - 9 metrics (energy / costs/ emissions)
  - 4 chart types
  - Recommendations
  - Portfolio Manager import

API Deployment

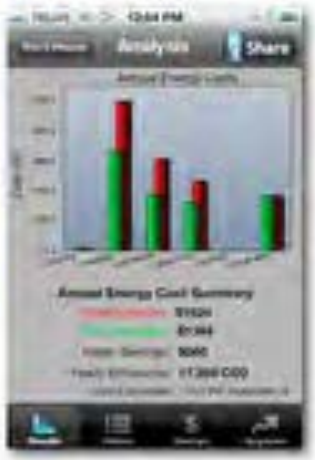
- About 300 entities have expressed interest in these APIs
- Within the first year, more than 50 entities (public and private) became users.
- Some launched products, others still working, others fell away
- More runs are generated via our APIs now than via our own GUIs

Microsoft *hohm*

- Who: Microsoft
- What: Consumer education

Mobile App: iViro

- Who: Envirolytics
- What: Consumer education & lead generator
  - Uses other apps to facilitate inputs
    - Compass for orientation
    - Camera + geometry for estimating wall heights and areas.

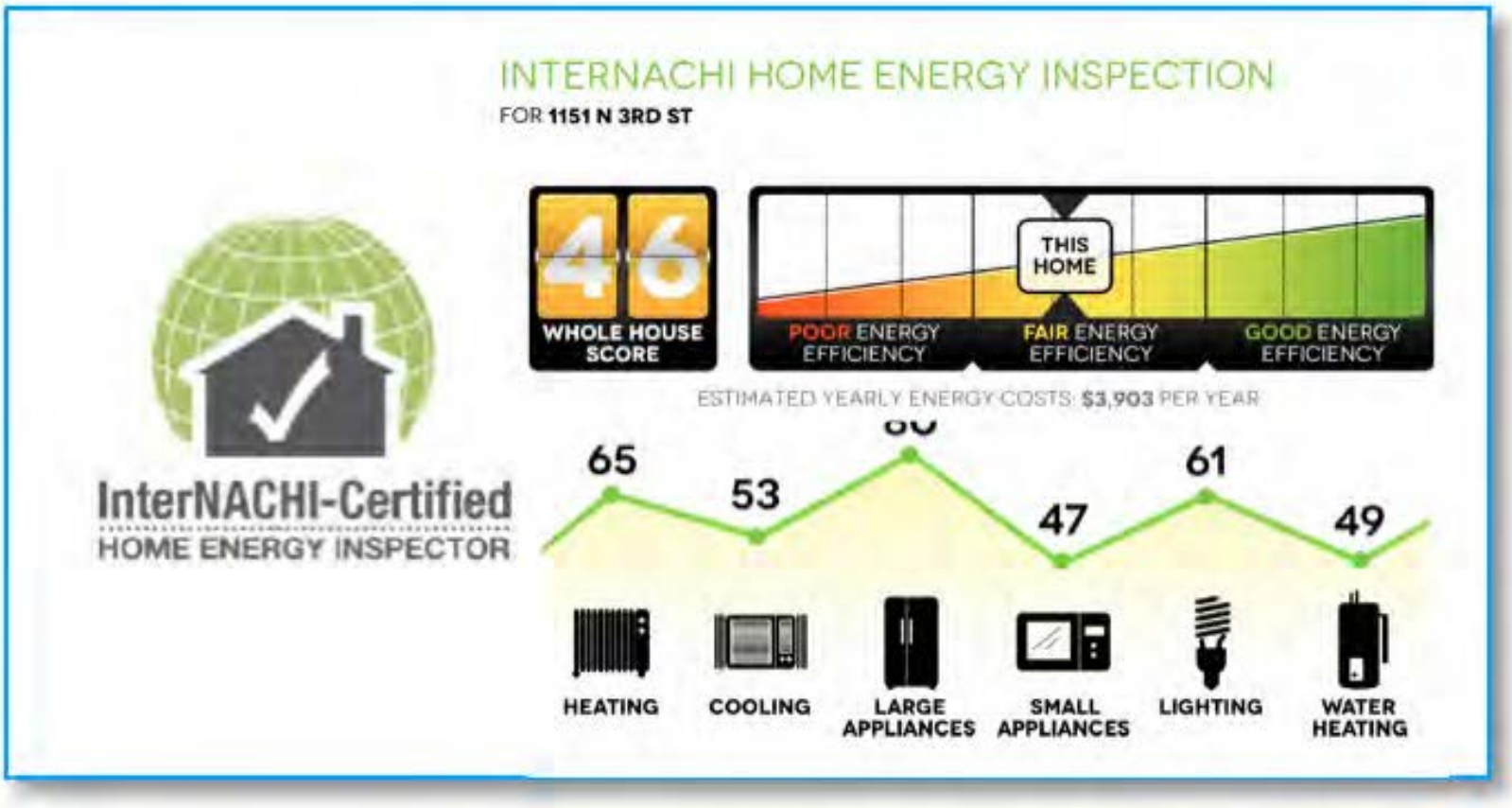


Consumer Tool

- Who: WattzOn
- What: Consumer education & lead generator

Home Inspector Tool

- Who: InterNACHI
- What: Home rating and tool for members

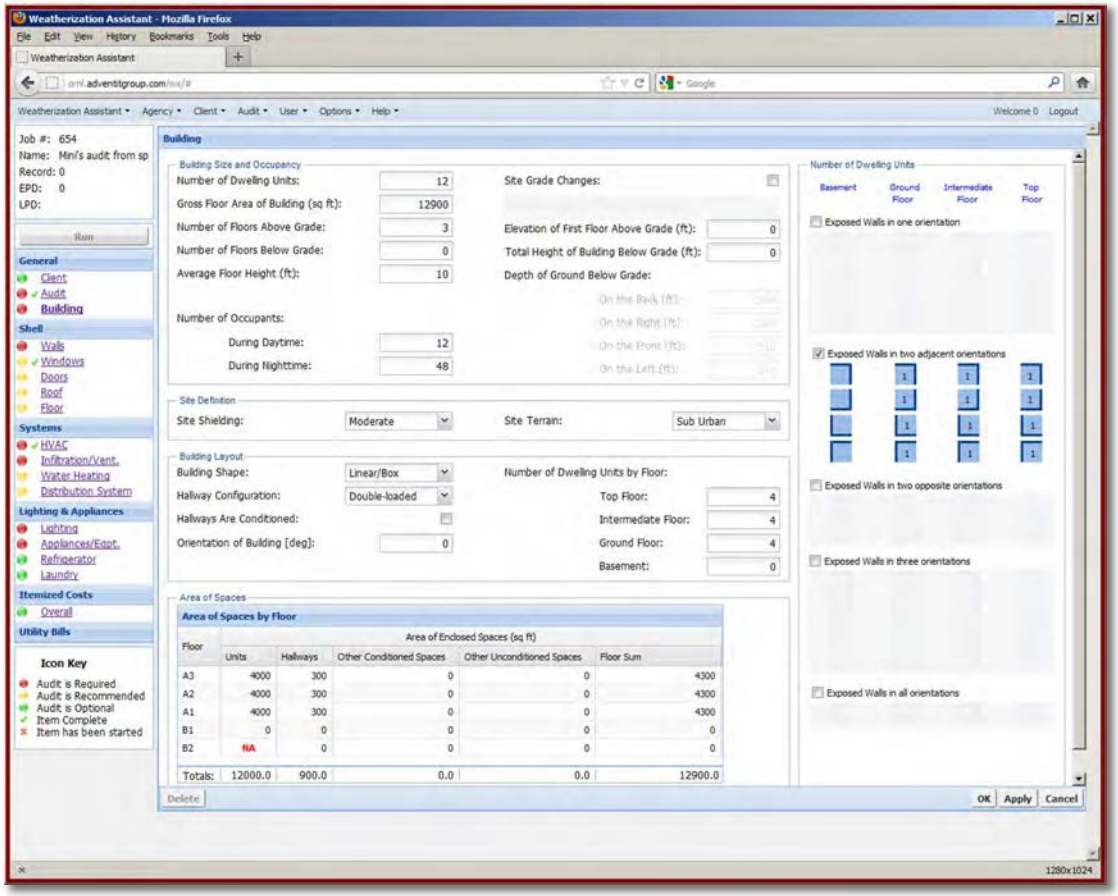


Home Energy Scoring Tool:
iOS (iPhone, iPad) Android (Phone, Galaxy Tab)

- Who: MNCEE
- What: Scores plus software for project mg't

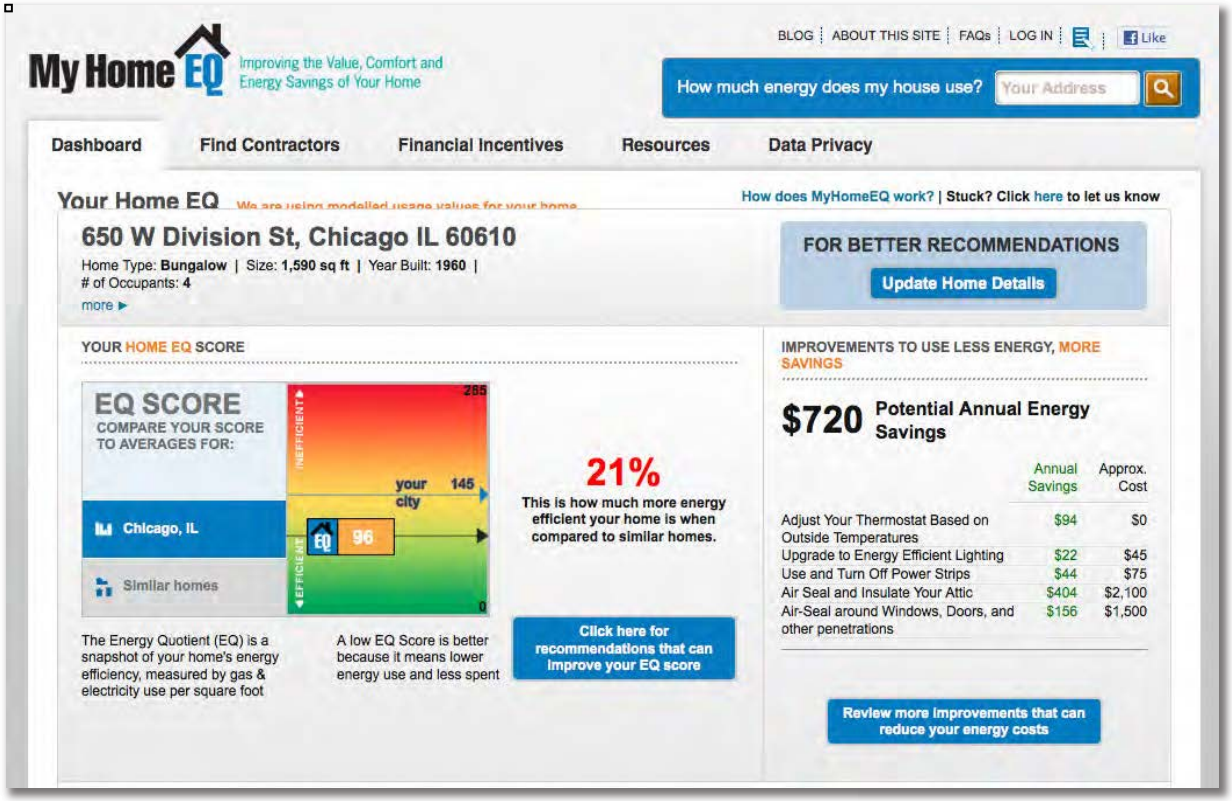
DOE WAP - MulTEA

- Who: ORNL
- What: Official tool for weatherization audits



MyHome-EQ

- Who: CNT spinoff
- What: Home rating + lead generation

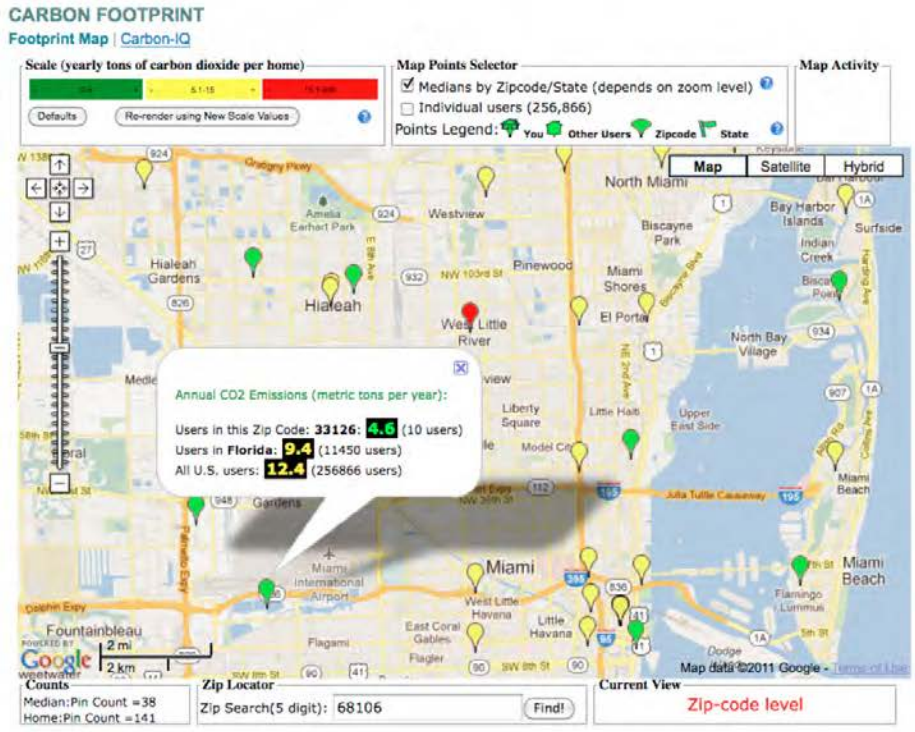


CoolCalifornia

- Who: California Air Resources Board
- What: Carbon calculators for homes and businesses

Future Directions

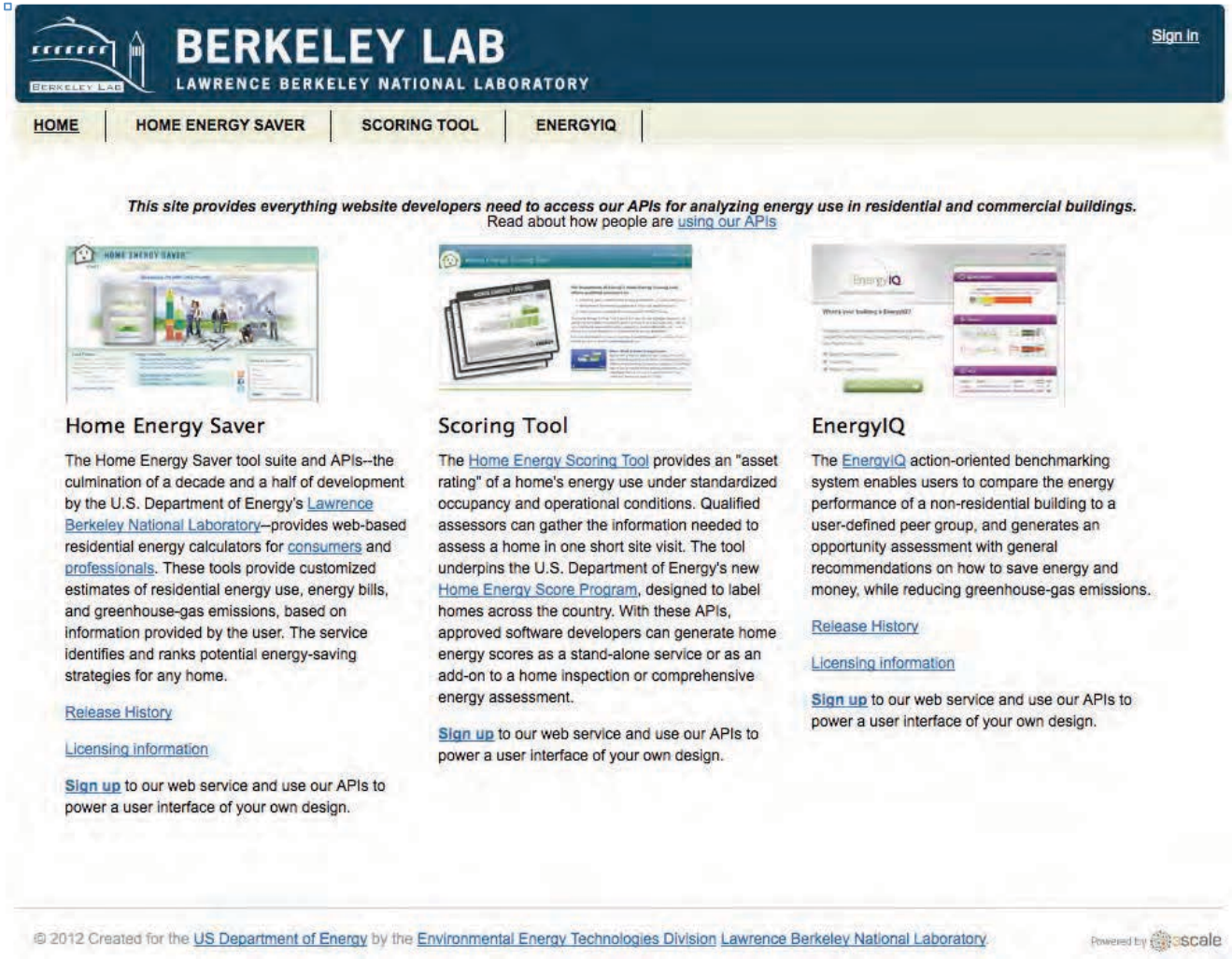
- Embedded *inside* devices (e.g. TSTATS)
- EMS integration
- Mash
- Etc…



Take-aways

- APIs are powerful new technologies for energy modeling
- Public R&D investment is enabling private-sector innovation in user-interfaces and delivery, eliminating bottlenecks, and (potentially) supporting back-end standardization
- The public sector is a good place for developing APIs
  - There's no money in it
  - Innovative mode of technology transfer
  - Non-proprietary (no particular product or fuel orientation)
  - Staying power
- Not for the faint of heart
  - Development is grueling
  - Users are demanding
  - Stakes are high (others depend on your service)
- Users are (understandably) fickle

https://developers.buildingsapi.lbl.gov/



emills@lbl.gov