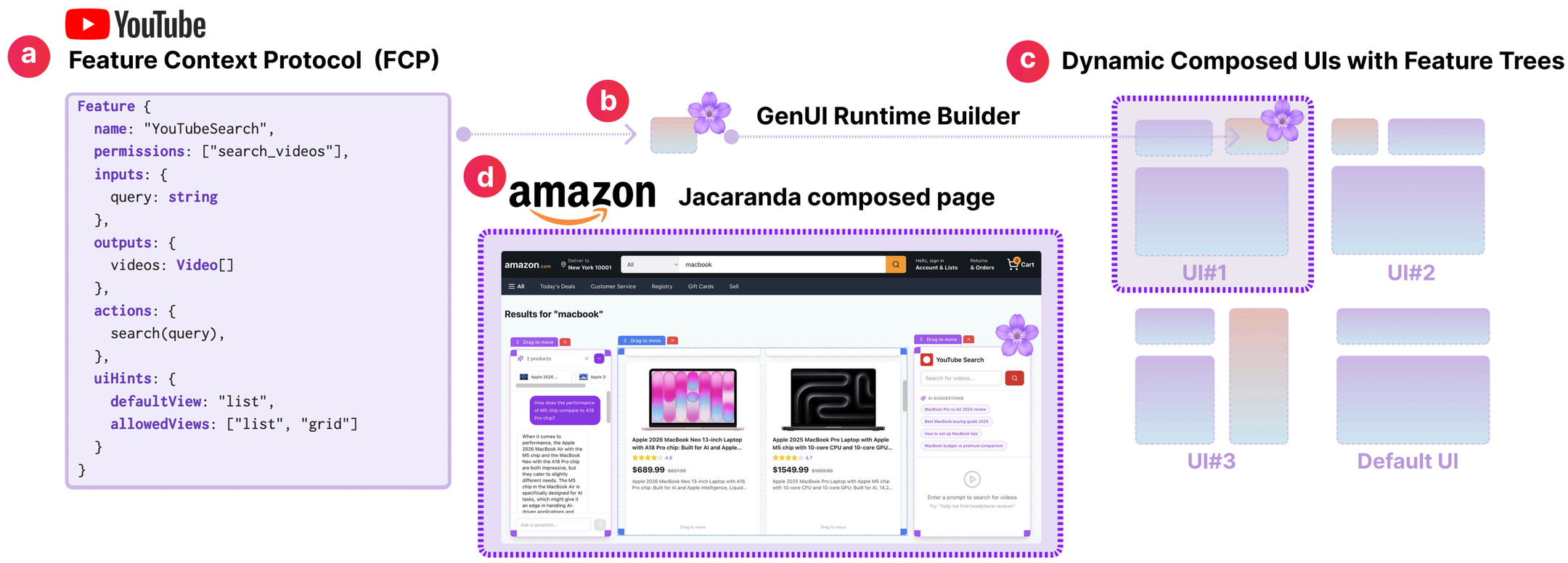
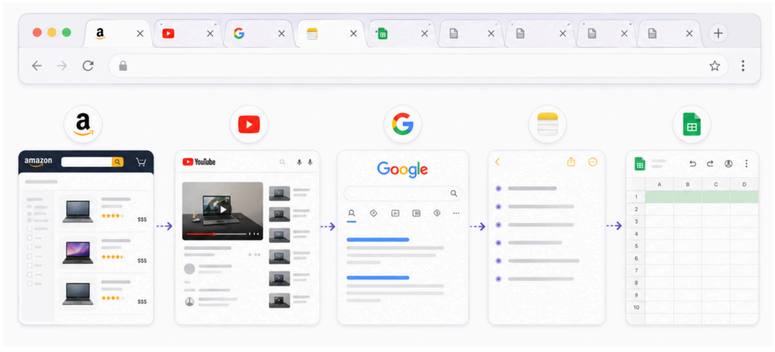
Jacaranda:
Grafting User-Editable Web Experiences Across Sites

Lets users reshape the web around their goals by grafting features from different sites into one editable feature-tree page.

Xinqi Zhang(xzhang22@scu.edu), **Mingyuan Zhong**(myzhong@cs.washington.edu), **Longjie Guo**(longjie@uw.edu), **Bryan Min**(bdmin@ucsd.edu), **Yuanhao Zhang**(yzhangiy@connect.ust.hk), **Kai Lukoff** (klukoff@scu.edu)



01 Motivation: The Current Web Is Fragmented



Today's web tasks rarely happen on one site.
A shopper comparing products may move between Amazon, YouTube reviews, search results and comparison tools. A student may watch a lecture, take notes, search references, and suppress distracting recommendations[1].

But web interfaces remain page-centered and platform-controlled. Features such as video search, captions, notes, and product reviews are locked inside separate sites, making them hard to extract, move, connect, or reuse[2].

02 Key Idea: From Pages to Feature Trees

Instead of treating websites as fixed destinations, as in traditional DOM-based web representation, Jacaranda takes inspiration from React's component tree and represents web experiences as **Feature-Tree Pages**: editable trees of reusable feature nodes.
In this design, a web page becomes a set of atomic features.
A feature node can be:

- a video player; a product search; a review summary;

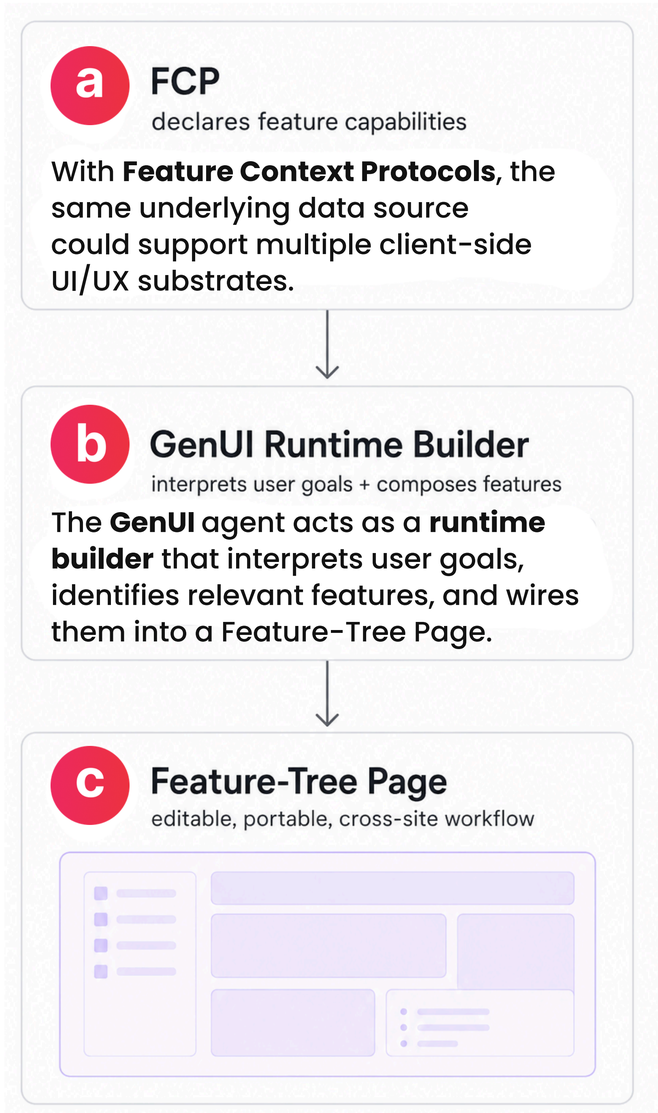
Users can move, reuse, and recombine these features across contexts.

03 System Model: GRAFT Generative Runtime Assembly of Feature Trees

Jacaranda is built around GRAFT, a system model for runtime-malleable browsing.

GRAFT has three goals:

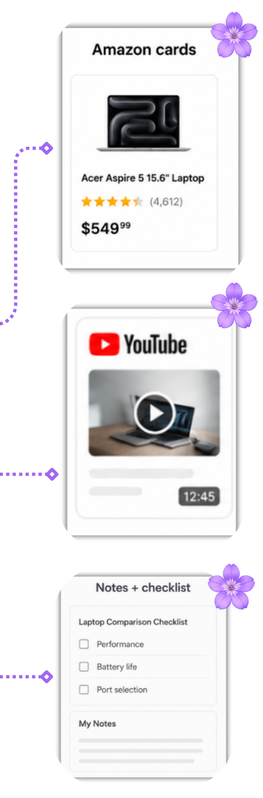
- **G1.** Decompose pages into meaningful feature units: Break fixed pages into reusable feature nodes.
- **G2.** Support user-defined composition: Let users describe goals and reshape workflows with GenUI assistance.
- **G3.** Enable cross-site portability: Allow features from different platforms to be reused in one personalized page.



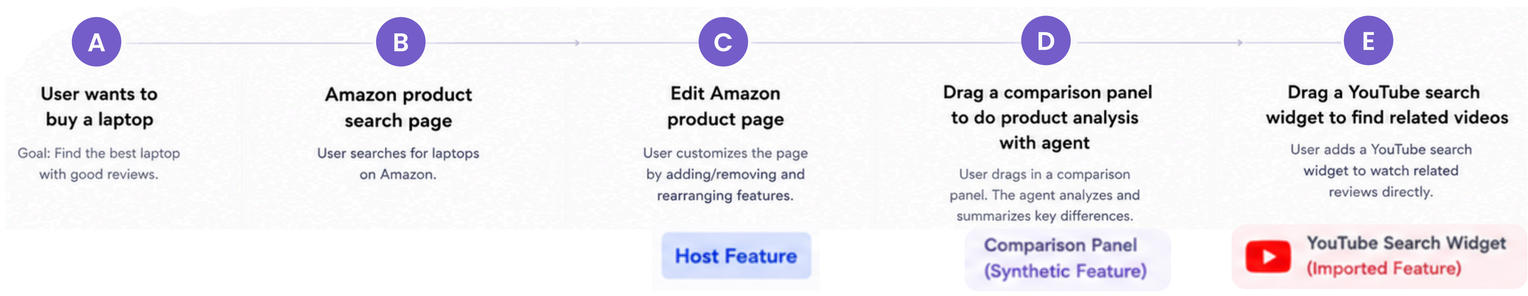
04 Feature Sources

Jacaranda composes features from three sources, taking Amazon.com as the host-site example:

- **Host Features**: Features from the current site, such as product cards, search results or captions.
- **Imported Features**: Features from other services, such as YouTube reviews inside an Amazon shopping workflow.
- **Synthetic Features**: Task-specific features generated for the user, such as notes, comparison board, summaries, or checklists.



05 Example Scenario: Shopping Workflow



06 Contribution

Jacaranda contributes:

1. A browser model for user-editable web experiences.
2. GRAFT, a system model for runtime feature composition.
3. A design framing for portable, cross-site feature reuse.

Future work will evaluate FCP-defined features, feature-tree pages in shopping, video learning, travel planning, and accessibility support. We will study how users express workflow goals, how much control they want over GenUI-generated changes, and whether feature-oriented browsing reduces tab switching and interface clutter.

References:
[1] Chang et al. 2021. When the Tab Comes Due: Challenges in the Cost Structure of Browser Tab Usage. CHI '21. https://doi.org/10.1145/3411764.3445585
[2] C. N. Klokmose, J. R. Eagan, S. Baader, W. Mackay, and M. Beaudouin-Lafon. 2015. Webstrates: Shareable Dynamic Media. In UIST '15, 280–290. https://doi.org/10.1145/2807442.2807446