Amazon Purchase Preferences and Defaults

**Goal**

The goal of this project is to first identify the existing problems associated with purchase preferences and understand the needs of businesses that use these preferences. Then create a unified set of tenets agreed upon by all businesses to guide a cooperative solution to these problems.

After this foundation has been created we will explore solutions as a group that can accommodate the individual team needs while addressing the customer position of being a cross-business shopper. We will attempt to see preferences as an ecosystem that has impact in aggregate and help to simplify the experience for customers while not inhibiting their behaviour. To do this we will explore both the lifecycle of the preference and the customer journey from low to high engagement.

**Definitions**

**Purchase preference**: A stored set of choices that allows a customer to complete a purchase. This could be only a single payment method for a digital purchase or a combination of items like a list of payment methods, a shipping address, a shipping method, and additional choices needed to ship a physical good.

**Payment preference**: A subset of a purchase preference containing only the payment method information.

**Customer present (CP) purchase**: A purchase initiated by a customer action like clicking a button. E.g. a digital purchase, a physical goods purchase, cart, 1-Click, etc.

**Customer not present (CNP) purchase:**  A purchase initiated by Amazon without a customer action. E.g. a subscription, membership renewal, etc.

**TSPC:**  True Single Page Checkout. The summary page for retail checkout where all choices are presented inline instead of spoking out to individual pages for payment and address selection.

**Eligibility:** Determining whether a given set of payment methods are valid for a given purchase. Eligibility is derived from a complex dimensional comparison between payment method(s), shipping address, shipping method, ASIN(s), business rules, regional rules, and many other factors. Since this calculation is so complex and ever-changing it is impossible to determine full eligibility in advance. Some businesses exist in mutually exclusive eligibility sets which makes no common instrument viable.

**History**

Currently there is no unified preference authority. Typically businesses collect, store, consume, and manage their preferences internally. The exception is the 1-Click preference which is shared between businesses. Since these businesses wholly own the storage of the preference it is impossible to view or display a customer’s preferences in a single location. When an issue occurs with a customer’s purchase information (e.g. an expired or lost card, a new home address, etc.) the customer must identify on their own what preferences exist across amazon, locate those individual management pages, and manually update the preference. This process is arduous and many preferences are overlooked causing purchase failures across the site, customer headache, and millions of lost revenue.

**Types of Purchase Preferences**

Purchase preferences can be loosely divided into two categories. **Customer Present** (CP) purchase preferences are convenience shortcuts to allow purchases to be made more quickly. This includes both 1-Click purchases and cart purchase defaults that prepopulate the TSPC and default notions like default payment methods and addresses. The purchase context can be known at the point the preference is used and the customer is available to respond to conditions that arise. The customer can also decide which context fits the purchase at hand and choose a preference that suits that context if multiple preferences are available.

**Customer Not Present** (CNP) purchase preferences are used to charge orders in the future and are future looking in nature. Customers are not available at the time of charge so errors must be handled out of band through emails and site messaging. Future looking language like “when available” must be used due to the fluctuating nature of balance instruments. These include all of our subscription businesses like Prime, Subscribe and Save, Periodicals, etc. CNPs tend to have more precise eligibility requirements and are tailored to the acting business or product. They may also include additional data like recurrence.

**The Lifecycle of a Preference**

Preferences exist in four main contexts and UI is generally built around these areas.

1. Creation. Preferences are generally created at the time of purchase as an artifact of purchase choices. These choices are then saved to a preference store. UI here is typically low or non-existent to avoid purchase friction.
2. Consumption. When using a preference for a CP purchase the user must interact with that preference. For digital, simply displaying “1-Click” on the button infers the preference use though not displayed and for retail additional controls are given to select between other preferences, to add a gift option, and potentially other features. For CNPs this is invisible.
3. Management. All preferences have a page to change the selected choices, either custom like Prime or shared like 1-Click.
4. Deletion. Currently preference deletion happens only as a side effect of another action, by either cancelling an order or deleting an address or instrument. Preferences would need to have a unique entity like a preference ID in order to be deleted directly.

**1-Click Description and Limitations**

The 1-Click purchase preference service is a very old service built on top of the identity team’s address book database and represents our primary shared CP preference. For each unique address ID we can associate a payment instrument (and other purchase information) with that address. The Your Address Book page is literally the same page as the 1-Click Management Page. When acting on a 1-Click preference you are acting on the address as the unique preference key. Deleting or editing an address erases your 1-Click settings. When adding a new address you have to manually recreate the preference.

* This requires an address first to instantiate the preference. However this model quickly became outdated once digital goods were offered on Amazon. It also limits each address to having only a single preference.
* Since 1-Click is the only shared preference service many businesses have used the 1-Click preference storage in ways unrelated to the 1-Click purchase method. The retail TSPC for example uses the 1-Click preference to store their default checkout choices. This creates a situation where changing your Kindle purchase settings also changes your TSPC defaults and vice versa.
* The 1-Click service can only store a single instrument ID and generally only supports credit cards. This prevents multi-tender choices like opting into or out of gift card (GC) usage. Currently businesses either always or never consume GC without a customer choice.
* The service offers only a single default which must be shared by all businesses. However since not all businesses accept the same payment methods, the default set is often not valid for all purchases. (e.g. a CN customer who wants to use COD for their physical goods and a different method for their digital purchases). The result is that a customer is presented with a fixup experience and must overwrite their default each time they purchase within that context.
* For businesses that strictly use the default for purchases (e.g. digital businesses) there is no way to switch between user purchase contexts. For example if a married couple share an Amazon account or if a customer wants to make both personal and work purchases. The only way to do this is to navigate to the 1-Click management page, change the default, return to the details page, complete the purchase, then return to the management page to change the default back. This still does not allow mixed purchase context for a single address.
* And finally, there are deficiencies in the implementation of the consumption portion of the service. Today we surface a button but no indication of what will be charged or where it will be shipped. This dramatically reduces customer confidence, especially after an initial problem. For retail consumption we expose a (poorly generated) name associated with each address but the name is difficult to interpret and provides little context. For digital only the button is shown. The lack of confirmation or display is the primary reason given for not using 1-Click. Followed by flexibility.
* The result is that even though we can show that 1-Click adoption increases purchasing the use of 1-Click is in a reverse relationship with customer engagement. The more and varied purchases you make, the less likely you are to use 1-Click except where required.

**The Single Default Unicorn**

On first thought (and often requested) is the very simple idea of having a single default purchase preference or singly a default payment method, shipping address, and shipping method for the site. For many US customers, providing a primary credit card and home address would cover the majority of cases. But for highly engaged users and in non-US regions, this model quickly breaks down.

Problems to solve for:

1. No single payment method can be used for every client. Less prominent in the US (for CC users) but becomes pronounced internationally. There are some clear divisions like digital vs. retail but the exclusions happen within these client groupings for other reasons. Since eligibility rules are derived from a complex matrix it is impossible to model these divisions in a determinate way. Any simplified model that uses a single instrument will necessarily encounter these conflicts with no resolution path. This is a major 1-Click escalation topic.
2. Engaged users sometimes have more complex usage patterns than a single instrument. For example personal vs. work purchases. Accounts shared with a spouse or devices for family members managed under a single account. And though these cases might not represent the largest consumer block it directly correlates with user engagement, individual spend, and higher lifetime value of users and will limit interaction and potential growth.
3. Some payment methods are by design limited in scope. For example COD or app based payment methods that only work on a particular device. Industry trends show an explosion of alternative payment methods which will become an increasing share of our business when taken together, though independently might only be a small share per method. These methods are necessarily excluded from a global preference.
4. There is no current default authority. Outside of 1-Click, default notions are handled client-by-client. A non-trivial effort would be required to get clients to honor a global default. Which in absence could cause customer confusion as to what scope the global default applies to. Businesses also would invariably create default overrides which further erodes the authority.
5. There are some political implications of tying businesses together under a single default. Businesses with larger market shares would steer the design and dominate feature roadmaps at the expense of new or smaller ones. This could potentially result in being shoehorned into a system primarily designed to accommodate retail checkout.

A common response to these challenges is to suppress errors and strive towards the US single card user. But given a decade of escalations around 1-Click, our plans for rapid regional expansion and onboarding of many new limited-scope payment methods, a flexible and extensible solution needs to be presented.

**Contextual Model - High Level Proposal**

1. Create a new preference solution that will allow new customers to establish a site wide purchase preference while making their first purchase with a minimum barrier.

1. Expose that preference and its contents site wide where possible.
2. Intelligently handle eligibility conflicts, expanding this simple preference only when required.

In this way simple users will have the simplest preference possible while more complex users will always have the simplest possible solution to their individual needs without being restricted.

Customers can easily access these preferences whenever checking out and have easy visibility and confidence of the outcome of any purchase.

Power users may easily expand upon these preferences in whichever ways make sense to them personally.

One preference concept that can account for the myriad of individual needs and ever-expanding payment methods with the lowest bar of entry and the simplest way to purchase.

**Suggested Tenets**

1. **Simple.** *Always strive for the minimum amount of preferences possible.*
2. **Growth.** *Allow for organic growth in the complexity of our user’s lives and behaviour.*
3. **Extensible.** *Be adaptable to new payment methods with variable scope and new regions with unexpected behaviours.*
4. **Transparent.** *Users should be clear and confident of the outcome of any purchase.*
5. **Don’t make me think.** *The task of understanding eligibility and creating/maintaining preferences should be natural and without thought and never distract from the user’s primary goal: to shop and make purchases.*
6. **Internally flexible.** *Our solution should accommodate different business and purchase models and allow for experience variation while maintaining a conceptually consistent object model for the customer. (i.e. it might look and behave a little different from client to client but as a user I still understand that this is the same preference and understand how to use and interact with it.)*

**Approach**

1. Create a universal preference authority. This authority would house all purchase preferences site-wide, allow cross-client visibility, enable site-wide fixup of payment and address changes, and allow for multi-tender payment preferences. Migrate businesses to this service.
2. Universal update. When a user makes a change to a payment method or address we can now scan their existing preferences and allow them to populate that change across the set and fix errors and conflicts. (e.g. deleting a card used by other preferences, replacing a lost card, adding a new card intended as the primary method, changing a primary address, etc.)

Shared 1-Click Preference (CP):

1. Move to a unique Preference ID instead of an address ID and make each preference element optional. In this way each business can collect only what their purchase requires, minimizing the bar to creation. Subsequent businesses that use the preference but need additional information (e.g. shipping address) can collect that information in flow to their purchase and update the preference.
2. Since the address is no longer the key, move to a user supplied name as the primary differentiation for shared preferences. Initial preference name can be generated but subsequent names should be user driven. This allows for highly variable user schemas with minimal effort or UI. Studies show user schemas are too various to quantify. See the existing Wishlist product as a good viable example of user naming and generic schema.
3. Reword the 1-Click preference phrasing as a “Purchase preference” or similar generalized language instead of the overly specific 1-Click preference language. Divorce the preference store (preference selection) from the 1-Click method of purchase (buy with 1-Click button). This way the new preference can be used for 1-Click purchases, Buy Now purchases, and on SPC as the same preference if desired.
4. Consistent UI. Create a standard consumption UI that allows selection of alternative preferences, a display of selected preference contents, and the ability to create a new pref while purchasing. I envision this as a widget that encapsulates the complex eligibility logic.
5. Upon first purchase, create a purchase preference and populate it across all CP consumption UIs. For most users this should effectively be a single global default. End of story. We can then use this preference to seed CNP purchases, turbo checkout default options, etc. A very low bar.
6. Handle conflicts at the point of purchase. If the current preference is ineligible, instead of overwriting a single default, we can fork the single preference into a second. This new preference could then be preselected in subsequent scenarios when needed. (e.g. a user chooses COD for their initial purchase which is populated into their purchase preference. That user then makes a digital book purchase. Instead of overwriting the original preference we create a second preference that the customer names, perhaps “Digital purchases”. When that customer encounters a similar conflict in an unrelated digital workflow, say MP3, the “Digital purchases” preference becomes preselected avoiding a second conflict. When that user purchases digital books again, the “Digital purchases” preference will be preselected. )

This will ensure the lowest possible number of preferences while accommodating any unavoidable eligibility conflicts.

They key here is to remove the single default concept and move to a context driven selection model to determine usage intelligently. When unknown, pick the best option, then remember the behaviour on return. This simple model can account for highly complex usage behaviour.

SPC/TSPC:

1. Expose preference selection on the retail summary page when more than one preference is available. For customers with only one preference, suppress the selection UI but when changes are made to the default, allow the customer to overwrite their single default (current) or save the new options as a secondary preference. If the user chooses to save as new, expose a field to name the new preference. In this way customers can save choices for different contexts like gifts sent to relatives, work purchases, etc.. This new preference then becomes available in 1-Click and other purchase paths.

Subscriptions and other CNPs:

1. Keep CNP preferences unique. User studies have shown that though users can create, conceptualize, and apply a single contextual preference across the site easily, they struggle to conceptualize eligibility errors that occur during universal fixup. CNPs tend to have the most exclusive eligibility and business specific decision trees. They also lack the customer present UI to handle errors. This requires all errors to be fixed at the point of update (we can’t defer complexity to the point of purchase) which tested very poorly. However moving CNPs to a universal authority allows for updates to be driven through both sets via an object model (acting on the instrument or address, etc.) which is much less complicated to present. When acting on the CNP preference as a whole, the user interacts with the business UI exclusively which tailors eligibility to that specific purchase context. Attempting a universal management UI is too complex to keep current and bottlenecks business growth.

Other areas:

1. Enable device level preferences. We can seed device preferences from a user’s primary preference but allowing for stand-alone device preferences has multiple advantages. a) Constrained devices like Xymal devices that do not have a web view can be isolated from the 1-Click ecosystem. Currently 1-Click must accommodate the lowest common denominator because all devices share the same single default. b) Devices can take advantage of new payment methods that are limited in scope to device-only like app based or hardware payment. c) Families that share a single Amazon account can have different payment methods on different devices. Spouses can make separate purchases, child accounts can be isolated from balance usage, work devices can charge work cards, etc.

**Contextual Expansion**

Building on this contextual model we can then add contextual queues to purchase preference entities. This added context would be very useful to tailoring purchase experiences within those contexts and informing charge logic and instrument selection/ordering. We already do this when adding an address with the optional field of work or home. If we expand that selection to include say Address for Gifting then we can infer that this address is less likely to use a work related card. Similarly knowing if a new payment method is Personal or For Work allows for betting wallet cycling behaviour. But this data collection would need to be tested against the barrier to entry (more form complexity).

Paradigm Shift

This approach requires a transition from the current single default to a less determinate contextual default. This shift is key to solving the single default fallacy. Two primary requirements arise:

Since there is no single default in this model, some ability to change the preference selection is required. In less space constrained UIs I imagine this as a dropdown of potential preferences with a display of the selected preference beneath. For space constrained UI a simple indicator (e.g. a small text line showing the pref name and card used with a change link, “Daniel’s preference: Visa \*\*\*\*1234 change” would be sufficient. But some access to alter the choice. Selection is not currently available for digital purchases and neither digital nor retail display the preference contents.

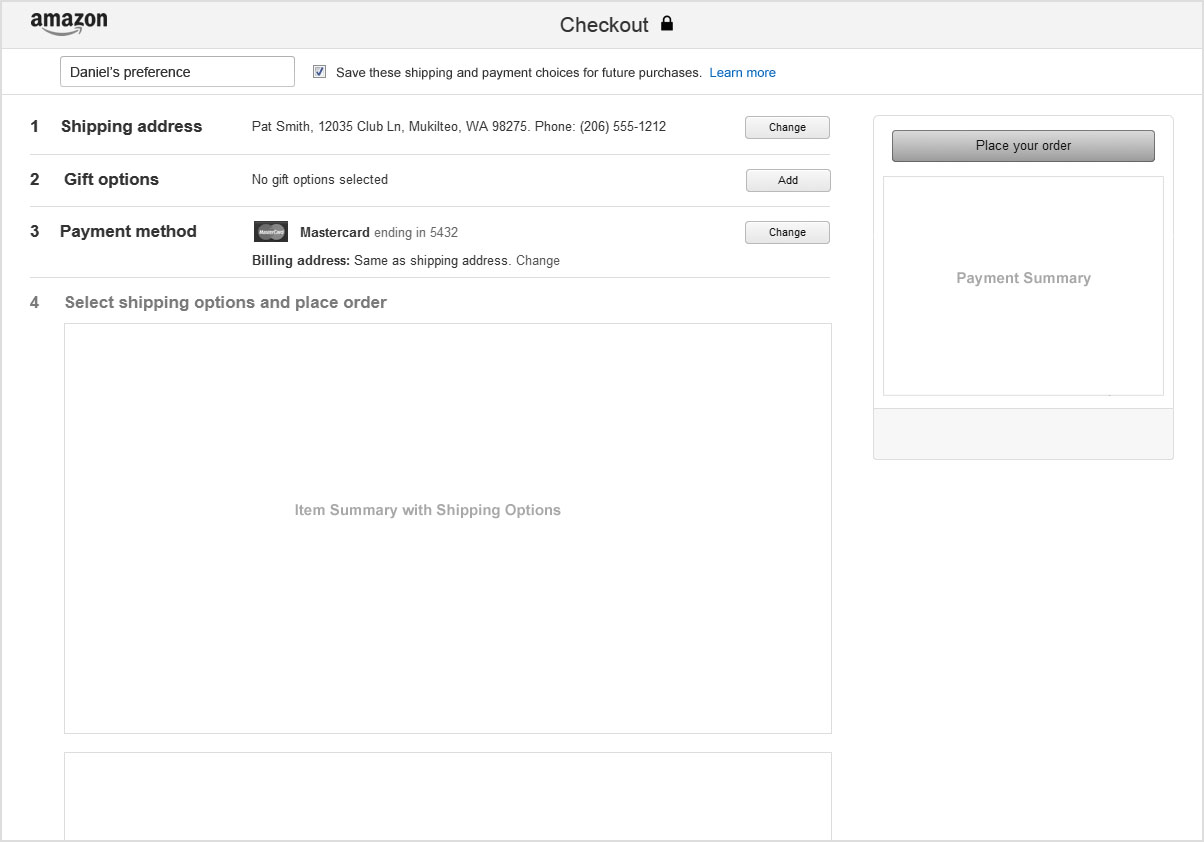
The second requirement is the ability to name a new preference when it’s created. User supplied names are essential (vs. generated names) to accommodate the highly variable user schemas encountered. In a survey conducted a few years ago, the structure in which users wanted to construct their purchase preferences was extremely variable. Any attempt to quantify these structures (i.e. provide a premade list of categories or divisions) necessarily limited customer choice or became unusably complex.

For a few examples: Personal vs. Work, Profile concepts like husband and wife or children and other family members (very common), preferences for gifting to various people, cost thresholds like over $X use this preference vs. under $x dollars use another, size thresholds e.g. “I ship to an Amazon locker which has size limits. I want one preference for large packages and one for small”, specific item or category level divisions like for groceries or baby items, EBT purchases, region of destination, divisions of user business context like different cost centers on a shared Amazon work purchasing account, nearly every person had a different model of how they wanted to structure their preferences and nearly all users in our study saw value in being able to do so even with only one credit card being used.

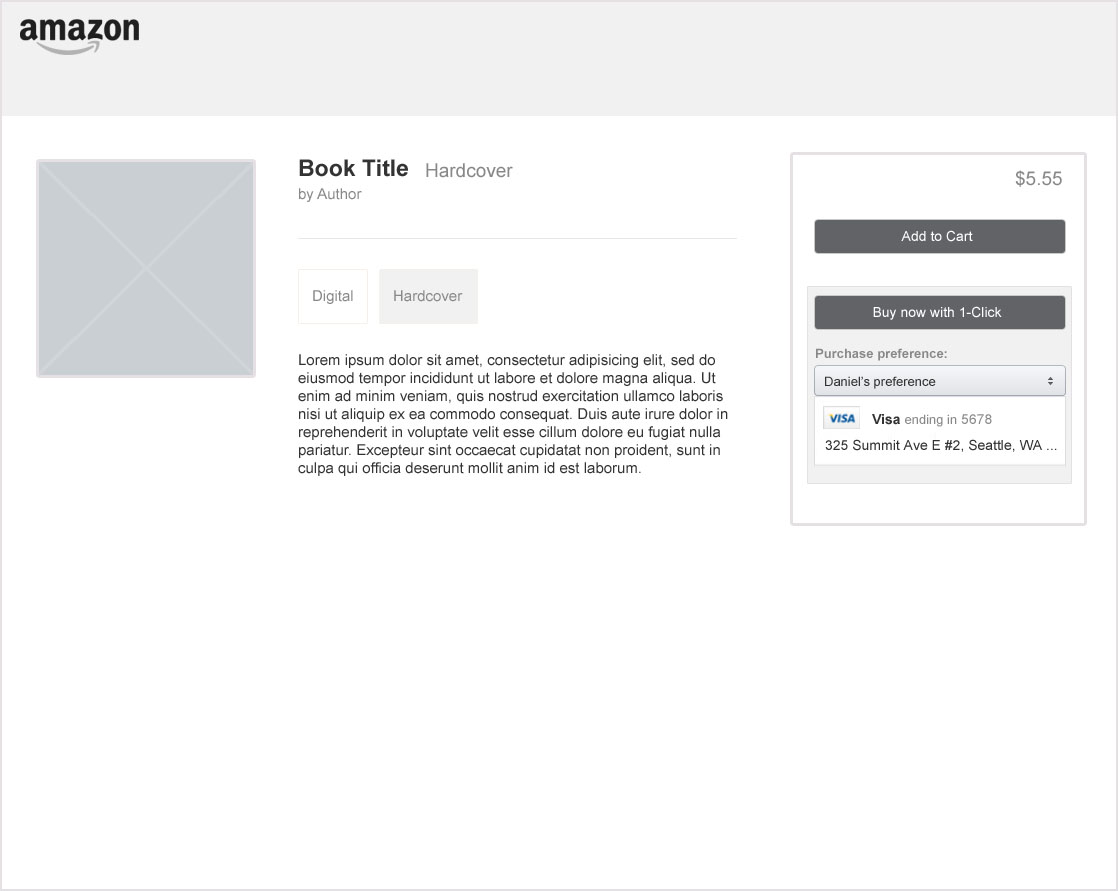
A much simpler approach is to allow the user to provide an arbitrary name. This name represents their personal schema division and can account for any division without limitation. Users will find ways to use our products in ways we can’t imagine. Let’s not get in their way.

A great example of this open schema paradigm can be seen in the Amazon wishlist product. The initial wishlist is presented as singular named by default but the option to create a new list is clear and forward. When a user creates a new list they are prompted to name the new list. This name is arbitrary and allows for complex list schemas to develop. I personally have over 40 lists ranging from Xmas shopping lists to multipart projects like a new computer build to topical lists like Kitchen Stuff and Music to profile based lists for friends and family to a public facing list that helps people shop for me. It’s the most flexible schema with the least complex UI and technology. Win win. And has been proven to work and be easily understood.

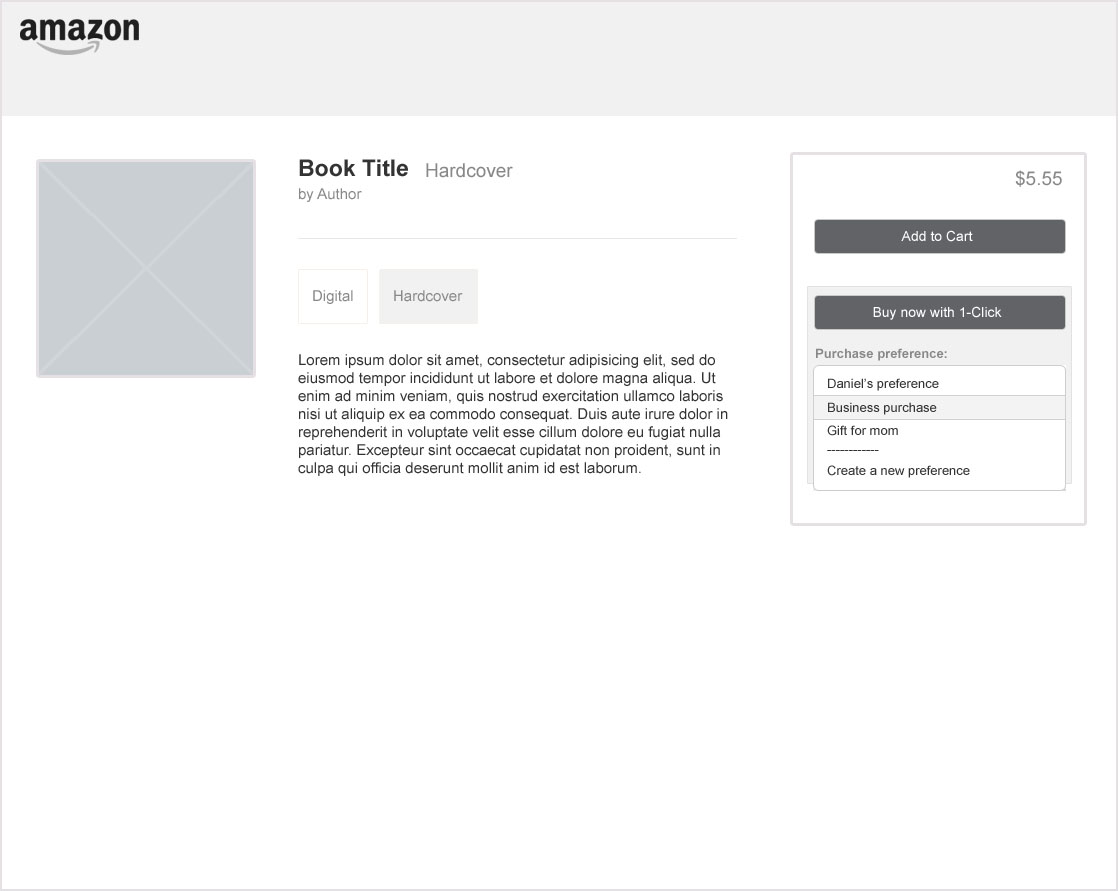
TSPC with naming concept



Sample retail consumption UI with selection and contents display



Selection dropdown with multiple preferences



1-Click UI refresh using naming paradigm. Here shown with current 1-Click wording and feature parity.

