

Peek-a-View: Smartphone Cover Interaction for Multi-tasking

Koeun Choi^{1,2}

Hyunjoo Song¹

Kyle Koh¹

Jinwook Bok¹

Jinwook Seo¹

¹Human-Computer Interaction Laboratory, Seoul National University, Seoul, Korea

²Software Center, Samsung Electronics, Seoul, Korea

{koeun.choi, hjsong, kyle, bok}@hcil.snu.ac.kr, jseo@snu.ac.kr

ABSTRACT

Most smartphones support multi-tasking with several means to switch between apps (e.g., a “recent apps” button or a “back” button). However, switching between apps is cumbersome when one has to do it frequently—for example, when notifications keep interrupting one’s current task. We introduce Peek-a-View, a fully transparent flipping screen cover that can reduce task switching overhead by providing an additional virtual screen space for subtasks. We assessed its feasibility in handling notifications. Upon receiving a notification, users can peek into the content of the notification without actually switching apps by slightly lifting the cover. If necessary, users can completely flip the cover to switch to the app that fired the notification. Two user studies showed that flipping and peeking interaction provided improved performance and proved to be useful for tasks that involve subtasks.

Author Keywords

Cover interaction; Mobile device; Task switching.

ACM Classification Keywords

H.5.2. Information interfaces and presentation (e.g., HCI): User Interfaces.

INTRODUCTION

Smartphones capable of performing multiple tasks at once are popular these days [1]. Multiple tasks include, but are not limited to, web browsing, video playback, social networking, and instant messaging (IM). Conventional multi-tasking in smartphones involved two steps to switch between apps: entering into app switching mode and selecting in the list of recent apps. Recent smartphone OSs (e.g., iOS 9 and Android Marshmallow) further introduced two techniques for multi-tasking: for permanent multi-tasking, one can split the screen space to display corresponding apps side-by-side; and for momentary multi-tasking, one can use a dedicated button to switch to a subtask and back out to the previous app

upon completion of a subtask. However, both techniques have limitations coming from inefficiency of a smaller screen space for each app [8] and repeated task switching.

A response to a notification is one of the main cases that prompts immediate switching between multiple apps [9]. Instant messaging, for example, asks for feedback on a notification when users are engaged in other activities, such as watching a video. Upon receiving a notification, one has to determine whether it is worth an immediate response, and then switch to the app, and return to the original task. Such task switching causes high overhead to users [7], but a small pop-up on an upper status bar often does not provide enough information (e.g., first few words of an instant message besides a name or a profile photo of the sender) to determine whether the switching is necessary. Thus, users have to do explicit actions before making a decision (e.g., switching to the app that fired the notification or enlarging the status bar for more information). Such interactions have three limitations in terms of supporting efficient multi-tasking: (1) it requires multiple interactions such as entering an app switching mode or the home screen; (2) using a back button does not restore the original state in the previous app immediately, whereas notifications usually provide a “deep link” within an app (e.g., a particular chatroom); (3) the use of a back button is irreversible that switching again to the app that raised the notification requires additional interaction.

To streamline the task switching process, we propose Peek-a-View, which utilizes a transparent flip cover to introduce an additional virtual screen in multi-tasking scenarios. Prior work on a smart device cover ranges from adopting it as an auxiliary input device with additional buttons on the cover [12] to using it as a secondary output for either display of relatively simple information [3] or full duplication of primary display contents with an e-ink panel [4]. Rendl et al. [10], on the other hand, introduced a thin transparent sensing

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org. CHI16, May 07 - 12, 2016, San Jose, CA, USA

Copyright is held by the owner/author(s). Publication rights licensed to ACM.

ACM 978-1-4503-3362-7/16/05\$15.00

DOI: <http://dx.doi.org/10.1145/2858036.2858426>

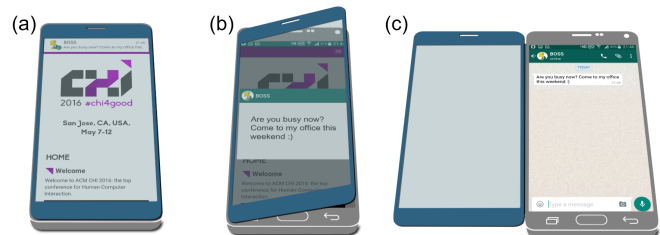


Figure 1. Three cover states:
(a) Closed – showing App-A, (b) Peeked – showing quick context of App-B, (c) Opened – showing App-B.

surface to be attached to a touch screen. It enriched interactions on the touch device by using deformation of the film as an additional interaction dimension. Previous researches also explored multi-layer interaction using additional mobile devices on the tabletop [13, 14]. For mobile devices, Cauchard et al. [2] used directional information acquired from an additional sensor to switch between virtual workspaces. As an alternative method, one could utilize the internal proximity/gyro sensors or swiping gesture to switch the workspaces, but the techniques could suffer from unstable grip or interfere with existing gesture mapping. Peek-a-View uses a readily available fully transparent flip cover that does not require any additional sensors. Due to its transparency, users can see and interact with the entire touch screen with the cover shut as if there is no additional interaction medium.

PEEK-A-VIEW

Peek-a-View adopts “virtual desktop” to address limited screen space for multi-tasking on mobile devices. Virtual desktop is a system used to expand virtual workspace beyond the physical limits of a screen’s display area. It allows the user to switch between different screen spaces on a PC [11]. Our system is composed of two virtual spaces that the user can interact with. Each virtual space can be switched by flipping the cover. It enables users to perform tasks on explicitly separate workspaces. In addition to the *closed* (Figure 1a) and *opened* (Figure 1c) state of the cover, the combination of proximity and light sensors enabled a new *peeked* (Figure 1b) state. We assigned two virtual workspaces provided by *opened* and *closed* state to two different apps, and introduced “peek” interaction to show a brief information of the app on secondary display. The system was implemented with built-in sensors (i.e., light sensor and proximity sensor) in the mobile phone. We defined each state with the combination of sensor values (i.e., *closed*: $proximity = 0, light < threshold$, *peeked*: $proximity = 0, light > threshold$, *opened*: $proximity > 0$).

FEASIBILITY STUDY OF COVER-CLOSED INTERACTION

A study was designed to (1) examine the affordance of a cover interface and (2) check which of the two workspaces (a covered screen, Figure 1a, or a bare screen, Figure 1c) is more suitable for the main virtual display, while keeping the other for subtasks.

Design and Procedure: We prepared short passages of writing over various topics (each with 152 ± 15 words). We extracted 18 passages from the Internet. After that, we measured their difficulties through an online-survey participated by 15 volunteers aged 20 to 38 years ($M = 30, SD = 4.59$) based on criteria made by Jo et al. [5]. We used six easy and one difficult passages from [5] for the difficulty baseline. After sorting the passages, we picked the 12 easiest passages out of the 18, which we combined with the six easy baseline passages to make a total of 18 passages for the experiment.

We investigated task switching scenarios using notifications:

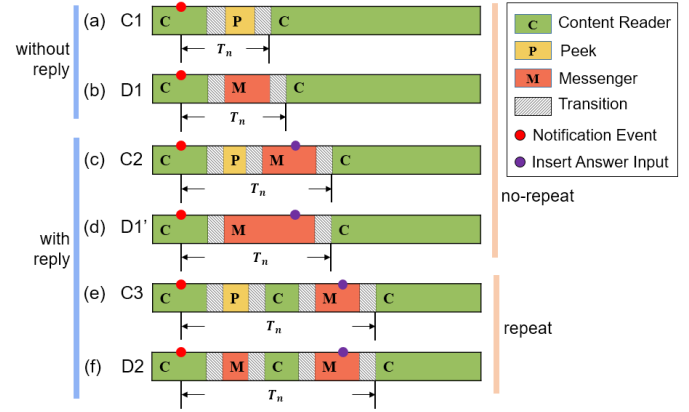


Figure 2. Response Task Switching Patterns. C1, C2, C3 means the pattern only using Peek-a-View. D1, D2 can be done with Peek-a-View (CO & CC) and Baseline (BS).

NT-N (No interaction): Messages that participants do not need to answer as in Figures 2a and 2b. (e.g., “You have a new message. Please ignore this message.”)

NT-O (One interaction): Messages that participants MUST reply to immediately as in Figures 2c and 2d. (e.g., “You have a new message. Please put an answer of $3+4=?$ ”)

NT-M (Multiple interactions): Messages that participants MUST reply to but they need to refer to the content from their current task as in Figures 2e and 2f. (e.g., “You have a new message. What is the character count in the first word of the 3rd line on a given passage?”)

All messages were over two lines long, and the first line deliberately showed the same text for all the types to make sure that participants were not able to figure out the type from the notifications without reading the entire message.

We built two apps: Content Reader and Messenger. Content Reader was composed of a text viewer and a “start” and “end” button. A pseudo-messaging app, which we designed after analyzing the top 5 most popular global mobile messenger apps (i.e., WhatsApp, QQMobile, WeChat, Skype, and Line) [15], delivered the notification messages. All five messengers required pressing a “back” key twice to exit the apps (once to leave a chatroom, and once again to exit from the applications). We excluded the Facebook Messenger and Viber because their popups block the viewport of the display unlike the others.

The participants performed the task with three interfaces: with a cover starting in flip-cover-closed state (CC); flip-cover-opened state (CO); and without a cover as a baseline (BS). BS includes the following existing task switching methods: using “back”; “home”; and “recent apps” buttons.

Each trial lasted for one minute. Only one notification, which appeared after 10 seconds into a trial, was received for each trial. All the three interfaces were performed six times (trials) ($3 \text{ types} \times 2 \text{ repeated}$), and 18 passages were assigned to each of the 18 trials using a balanced Latin square. Once the participants pressed a “start” button, they were told to read a

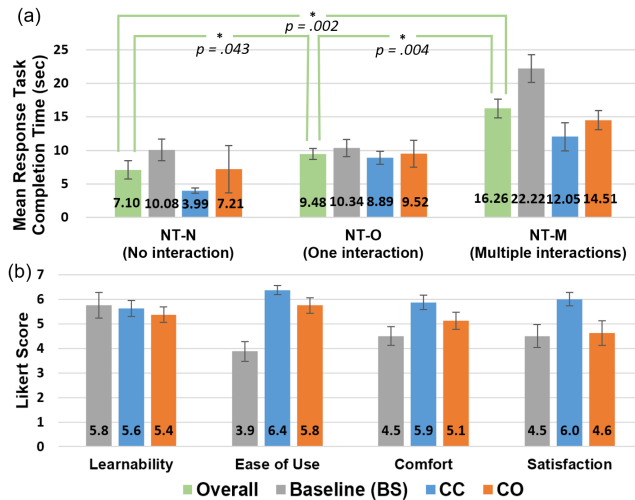


Figure 3. (a) Mean response task completion time and (b) mean questionnaire results. Error bars denote standard error.

passage. We asked them to check the message upon receiving them, take appropriate action, and return to the original task to continue reading the passage. When they pressed the “end” button, a question popped up asking about the content of the passage, providing three choices (a correct answer, a wrong answer, and “I don’t know”). Upon selection of a choice, the next trial started.

A three-minute training session was conducted for each method. After all the trials, we collected feedback using a seven-point Likert scale questionnaire (1=strongly disagree, 7=strongly agree), followed by an interview. A questionnaire asked for learnability, ease of use, comfort level, and satisfaction.

Apparatus: Our prototype was built on a Samsung Galaxy Note 4 (153.5 × 78.6 × 8.5 mm) running Android 5.0. A fully transparent tempered glass flip cover preserved the clarity of the screen.

Participants: Eight volunteers (two females) from a university participated in the user study. They were aged 20 to 38 years ($M = 29$, $SD = 4.70$) and have used Samsung Galaxy series smartphones for over six months.

Results

We analyzed the results using two-way RM-ANOVA and found significant main effects on response task completion time by interface ($F_{2,14} = 5.8$, $p = .015$) and notification type ($F_{2,14} = 19.728$, $p < .001$) (Figure 3a). There were also significant interactions on interface and notification type ($F_{4,28} = 3.224$, $p = .027$). Post-hoc tests using Tukey’s pairwise comparisons showed a significantly shorter ($p = .008$) response time (sec) using CC ($M = 8.215$, $SD = 0.696$) than BS ($M = 14.214$, $SD = 1.365$). As predicted, there were significant differences among NT-N, NT-O, and NT-M by Tukey’s pairwise comparisons (Figure 3a).

We also analyzed the questionnaire results (Figure 3b) using two-way RM-ANOVA. Easiness to use and satisfaction were significantly different between CC and BS ($F_{2,14} = 22.980$, p

$< .001$ and $F_{2,14} = 4.057$, $p = .041$, respectively), whereas there were no significant effects on learnability and comfort level. Post-hoc tests using Tukey showed that ease of use and comfort level for CC were significantly higher than BS ($p = .020$).

The study showed the better speed of cover interface than others, especially for CC. Moreover, there were meaningful results for CC on user preferences.

IN-DEPTH COMPARATIVE USER STUDY

Based on the findings from the previous study, we conducted a follow-up user study to further evaluate the efficacy of Peek-a-View using CC in supporting task switching.

Method: We prepared 18 passages and a device identical to the one in the previous study. We used a within-subjects design with two independent variables: interface (CC and BS) and notification type (NT-N, NT-O, and NT-M). Each participant performed three trials per condition, which resulted in $2 \times 3 \times 3 = 18$ trials per subject. For each trial, each participant was asked to do same task as the previous study. We measured the task completion time in each phase of the task. After the experiment, the participants filled out a questionnaire based on a modified SUS (System Usability Scale) [6] for subjective evaluation.

Participants: We recruited 18 participants (five females) from a university and a company. The participants were aged 20 to 34 years ($M = 26$, $SD = 4.67$), and all of them had experience with the Galaxy series smartphones for more than six months. They were compensated about US\$10.

Results

We first evaluated efficiency in terms of task completion time and then analyzed the usability of the proposed interface using questionnaires. Since each task included primary and secondary tasks and transitions between the two, we measured every individual time span including the transition time. Then, we defined response task completion time, and further scrutinized the subsequence of the task. We used two-way RM-ANOVA in temporal analyses.

Overall task completion time (sec): Overall task completion time was defined as the time span from the beginning till the end of each task. It included the time to finish both reading and response task. CC was faster ($M = 45.6$ and $M = 47.9$, respectively), but the difference was not significant.

Response task completion time (sec): Response task completion time was defined as the time span from the arrival of the notification to the completion of transition after the response. We analyzed response task completion time (T_n in Figure 4) and found a significant main effect of interface ($F_{1,17} = 112.37$, $p < .001$), and notification type ($F_{1,24,21,12} = 15.98$, $p < .001$). There was a significant interface × notification type interaction effect ($F_{1,25,21,25} = 15.98$, $p < .001$). For all the three notification types, CC was significantly faster than BS (Figure 4), and post-hoc tests using Bonferroni’s pairwise comparisons showed significant

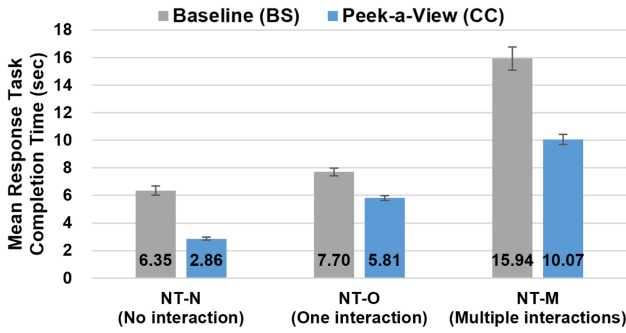


Figure 4. Mean response task completion. Error bars denote standard error.

differences between all the three notification type pairs ($p < .001$). However, the difference between CC and BS was relatively smaller in NT-O resulting in the significant interaction effect.

Notification recognition time (sec): Since each response could be divided into recognition, preparation, and text entry, we further scrutinized the elapsed time for each action. We excluded NT-O notification for instant response as the latter two actions could not be distinguished when one responded to the notification immediately. There was no significant difference for preparation and text entry, but recognition time using CC was significantly faster ($F_{1,17} = 119.06$, $p < .001$) for both NT-N ($M = 1.16$ and $M = 2.36$, respectively) and NT-M ($M = 1.97$ and $M = 2.86$, respectively).

Transition time (sec): Besides the elapsed time on both apps, we also measured transition time between them. We found a significant difference ($F_{1,17} = 106.135$, $p < .001$) between CC and BS ($M = 0.064$ and $M = 0.763$, respectively). Notification types also showed a significant difference ($F_{1,44,24.43} = 19.642$, $p < .001$) and post-hoc tests using Bonferroni's pairwise comparisons showed significant differences between NT-N and NT-M ($M = 0.340$ and $M = 0.583$, respectively, $p < .001$) and between NT-O and NT-M ($M = 0.317$ and $M = 0.583$, respectively, $p = .001$).

Peeking: [16] showed the average reading speed of text on a screen was 180 words per minute. Base on this, we speculated that it would take at least 333ms to read one word. The peeked state was used for 88.8% ($n = 142$, $M = 1.386$ sec, $SD = 0.611$) of all trials when we counted the occasion participants stayed in peeked state for longer than 333ms.

Usability: We analyzed the SUS rating for two interfaces using a paired-samples t -test. There was a significant difference ($t_{17} = -4.76$, $p < .001$) between CC ($M = 80.833$, $SD = 11.568$) and BS ($M = 67.083$, $SD = 11.056$).

DISCUSSION

Analysis on time spent for each action revealed that the superior efficiency of Peek-a-View came from faster recognition of notification content and faster transition time between apps. In the conventional condition, the participants showed mixed usage of the “back” and “recent apps” buttons for transition and barely used the “home” button to switch

apps. While conventional methods involve two interactions (i.e., one for entering a switch mode and the other for selecting an app), Peek-a-View needs only one physical interaction. It provides peek capability to check the content quickly with a single hand. However, more than half of the participants used both hands to flip the cover completely. It depicts the difficulty of single-handed flipping, but the interaction usually involves performing intensive tasks such as keyboard typing and non-casual browsing, in which case the use of both hands is advantageous. It showed strength in frequent app switching with its reversibility compared with the use of “back” button. In particular, the participants had to use the “recent apps” button for NT-M because of the need for multiple switching. Along with efficiency, analysis on the SUS rating also showed that the flip cover interface was more preferred by the participants. It could be attributed to familiarity with the flip cover as majority of the participants (77%, $n = 14$) have used or are using the cover.

Except for one participant who did not actively use the *peeked* state, the utilization was at 92% ($n = 141$). P7 and P14 said it was useful for NT-N and NT-M, but not so much for NT-O because they had to open the cover eventually to respond. NT-N and NT-M would benefit from the *peeked* state, because a natural follow-up action for both types would be to come back to the original task after checking the message in the *peeked* state, whereas NT-O would require entering the subtask. However, since the *peeked* state is the transitional state into the *opened* state considering the movement of the cover, the penalty would be at its minimal.

In our prototype, we relied on proximity and light sensor to detect the state of the flip cover. It showed feasibility even with the existing features of smartphones. While the number of states is limited to three in the current implementation, the three states were proved to be sufficient for multi-tasking between two apps. Also, P8 and P11 said that they wanted to see more notifications than just the most recent one. A combination of additional magnetic and other sensors can be used to detect the angle of the cover to achieve finer granularity and introduce a higher number of states for task switching between more than two applications. Furthermore, it can enrich the type of interactions more than a “peek”.

CONCLUSION

In this paper, we described and evaluated Peek-a-View, a flip-cover-based task switching interface. The proposed method outperformed the conventional button-based approach in terms of task completion time. It was also preferred by the participants based on subjective responses. Peek-a-View functions easily without disturbing standard features and is directly applicable to current devices.

ACKNOWLEDGEMENTS

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. NRF-2014R1A2A2A03006998). The ICT at Seoul National University provided the research facilities for this study.

REFERENCES

1. Matthias Böhmer, Brent Hecht, Johannes Schöning, Antonio Krüger, and Gernot Bauer. 2011. Falling asleep with Angry Birds, Facebook and Kindle: A large scale study on mobile application usage. In *Proceedings of the conference on Human-Computer Interaction with Mobile Devices and Services* (MobileHCI '11), 47-56.
<http://dx.doi.org/10.1145/2037373.2037383>
2. Jessica Cauchard, Markus Löchtefeld, Mike Fraser, Antonio Krüger, and Sriram Subramanian. 2012. m+pSpaces: virtual workspaces in the spatially-aware mobile environment. In *Proceedings of the conference on Human-Computer Interaction with Mobile Devices and Services* (MobileHCI '12).
<http://dx.doi.org/10.1145/2371574.2371601>
3. Ayumi Fukuchi, Koji Tsukada, and Itiro Siio. 2013. AwareCover: Interactive cover of the smartphone for awareness sharing. In *Universal Access in Human-Computer Interaction: Applications and Services for Quality of Life*, Constantine Stephanidis and Margherita Antona (eds.). Springer Berlin Heidelberg, New York, 620–625.
http://dx.doi.org/10.1007/978-3-642-39194-1_71
4. Gizmag. Prototype mobile phone covers foreshadow new wave of E Ink displays, Retrieved August 6, 2015 from <http://www.gizmag.com/e-ink-mobile-phone-covers/29063/>
5. Jaemin Jo, Bohyoung Kim, and Jinwook Seo. 2015. EyeBookmark: Assisting recovery from interruption during reading. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '15), 2963–2966.
<http://doi.org/10.1145/2702123.2702340>
6. Philip Kortum and Mary Sorber. 2015. Measuring the usability of mobile applications for phones and tablets. *International Journal of Human-Computer Interaction* 31, 8: 518-529.
<http://doi.org/10.1080/10447318.2015.1064658>
7. Luis Leiva, Matthias Böhmer, Sven Gehring, and Antonio Krüger. 2012. Back to the app: the costs of mobile application interruptions. In *Proceedings of the conference on Human-computer interaction with mobile devices and services* (MobileHCI'12).
<http://dx.doi.org/10.1145/2371574.2371617>
8. Nielsen Norman Group. Multitasking on Mobile Devices, Retrieved August 5, 2015 from <http://www.nngroup.com/articles/multitasking-mobile>
9. Martin Pielot, Karen Church, and Rodrigo de Oliveira. 2014. An in-situ study of mobile phone notifications. In *Proceedings of the conference on Human-Computer Interaction with Mobile Devices and Services* (MobileHCI '14), 233-242.
<http://dx.doi.org/10.1145/2628363.2628364>
10. Christian Rendl, David Kim, Sean Fanello, Patrick Parzer, Christoph Rhemann, Jonathan Taylor, Martin Zirkel, Gregor Scheipl, Thomas Rothländer, Michael Haller, and Shahram Izadi. 2014. FlexSense: A transparent self-sensing deformable surface. In *Proceedings of the symposium on User interface software and technology* (UIST '14), 129–138.
<http://doi.org/10.1145/2642918.2647405>
11. Meredith Ringel. 2003. When one isn't enough. In *Proceedings of the SIGCHI '03 Extended Abstracts on Human Factors in Computing Systems* (CHI '03), 762-763. <http://doi.org/10.1145/765891.765976>
12. Fletcher R. Rothkopf. 2012. Cover attachment with display, Patent No. US20120194448, Filed August 11, 2011, issued August 2, 2012.
13. Martin Spindler, Marcel Martsch, and Raimund Dachsel. 2012. Going Beyond the Surface: Studying Multi-Layer Interaction Above the Tabletop. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI'12), 1277-1286.
<http://dx.doi.org/10.1145/2207676.2208583>
14. Martin Spindler, Wolfgang Büschel, and Raimund Dachsel. 2012. Use Your Head: Tangible Windows for 3D Information Spaces in a Tabletop Environment. In *Proceedings of the 2012 ACM international conference on Interactive tabletops and surfaces* (ITS'12), 245-254. <http://dx.doi.org/10.1145/2396636.2396674>
15. Statista. Most used instant messaging apps 2015, Retrieved September 10, 2015 from <http://www.statista.com/statistics/258749/most-popular-global-mobile-messenger-apps/>
16. Martina Ziefle. 1998. Effects of display resolution on visual performance. *The Journal of the Human Factors and Ergonomics Society* 40, 4: 554-568.
<http://dx.doi.org/10.1518/001872098779649355>