

Reliable Interface to *GPT* via *Gmail*

Copyright (C) 2023 Hee Shin

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of *MERCHANTABILITY* or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see < <https://www.gnu.org/licenses/> > .

This specification models the coordination between the task that retrieves e-mails and the completion task.

CONSTANTS *Signals*, the set of signals notifying the completion task
Emails the set of incoming e-mails

VARIABLES *Channel*, the synchronisation channel between the tasks
CompletionQueue, the queue of completed e-mails
EmailQueue, the queue of incoming e-mails
QueueChanged whether *EmailQueue* has changed by the *EnqueueEmail* action

$vars \triangleq \langle Channel, CompletionQueue, EmailQueue, QueueChanged \rangle$

$AllQueues \triangleq CompletionQueue \cup EmailQueue$

Set of *Emails* found in any queue.

$UnqueuedEmails \triangleq Emails \setminus AllQueues$

Set of *Emails* not found in any queue.

$TypeOK \triangleq \wedge Channel \in SUBSET\ Signals$
 $\wedge CompletionQueue \in SUBSET\ Emails$
 $\wedge EmailQueue \in SUBSET\ Emails$
 $\wedge QueueChanged \in BOOLEAN$

$AllEmailsDelivered \triangleq \Diamond \Box (Emails = CompletionQueue)$

Temporal property that all e-mails are eventually delivered.

$SendSignal \triangleq Channel' = Channel \cup \{CHOOSE\ signal \in Signals : signal \notin Channel\}$

Inserts a signal in the channel to signal the completion task of an incoming *email*.

$ReceiveSignal \triangleq Channel' = Channel \setminus \{CHOOSE\ signal \in Channel : TRUE\}$

Removes a signal from the channel.

$EnqueueEmail \triangleq$

Enqueues an incoming e-mail in the *EmailQueue*.

$$\begin{aligned}
& \wedge QueueChanged = FALSE \\
& \wedge \exists email \in UnqueuedEmails : \\
& \quad \wedge EmailQueue' = EmailQueue \cup \{email\} \\
& \quad \wedge QueueChanged' = TRUE \\
& \wedge UNCHANGED \langle Channel, CompletionQueue \rangle
\end{aligned}$$

SignalCompletionTask \triangleq

Signals the completion task of an incoming e-mail.

$$\begin{aligned}
& \wedge QueueChanged = TRUE \\
& \wedge QueueChanged' = FALSE \\
& \wedge SendSignal \\
& \wedge UNCHANGED \langle CompletionQueue, EmailQueue \rangle
\end{aligned}$$

CompleteEmail \triangleq

Have an e-mail in *EmailQueue* completed when signaled.

$$\begin{aligned}
& \wedge Channel \neq \{\} \quad \text{EmailQueue has signaled the completion task.} \\
& \wedge \exists email \in EmailQueue : \\
& \quad \wedge CompletionQueue' = CompletionQueue \cup \{email\} \\
& \quad \wedge EmailQueue' = EmailQueue \setminus \{email\} \\
& \quad \wedge ReceiveSignal \\
& \wedge UNCHANGED \langle QueueChanged \rangle
\end{aligned}$$

Init \triangleq $\wedge Channel = \{\}$
 $\wedge CompletionQueue = \{\}$
 $\wedge EmailQueue = \{\}$
 $\wedge QueueChanged = FALSE$

Next \triangleq $\vee EnqueueEmail$
 $\vee SignalCompletionTask$
 $\vee CompleteEmail$
 $\vee UNCHANGED vars$

Spec $\triangleq Init \wedge \Box[Next]_{vars} \wedge WF_{vars}(Next)$

THEOREM $Spec \Rightarrow \Box TypeOK \wedge AllEmailsDelivered$

\ * Modification History
\ * Last modified Mon May 15 14:52:30 KST 2023 by hcs
\ * Created Sun May 14 11:53:48 KST 2023 by hcs