Zabbix Documentation 3.0

1.8 2.0 2.2 2.4
3.0
3.2 3.4

1 Macros supported by location

Overview

The table contains a complete list of macros supported by Zabbix.

Macro	Supported in	Description
{ACTION.ID}	→ Notifications and commands → Internal notifications	*Numeric ID of the triggered action.* Supported since 2.2.0.
{ACTION.NAME}	→ Notifications and commands → Internal notifications	*Name of the triggered action.* Supported since 2.2.0.
{ALERT.MESSAGE}	→ Alert script parameters	*'Default message' value from action configuration.* Supported since 3.0.0.
{ALERT.SENDTO}	→ Alert script parameters	*'Send to' value from user media configuration.* Supported since 3.0.0.
{ALERT.SUBJECT}	→ Alert script parameters	*'Default subject' value from action configuration.* Supported since 3.0.0.
{DATE}	→ Notifications and commands → Internal notifications	*Current date in yyyy.mm.dd. format.*
{DISCOVERY.DEVICE.IPADDRESS}	→ Discovery notifications	*IP address of the discovered device.* Available always, does not depend on host being added.
{DISCOVERY.DEVICE.DNS}	→ Discovery notifications	*DNS name of the discovered device.* Available always, does not depend on host being added.
{DISCOVERY.DEVICE.STATUS}	→ Discovery notifications	*Status of the discovered device*: can be either UP or DOWN.
{DISCOVERY.DEVICE.UPTIME}	→ Discovery notifications	*Time since the last change of discovery status for a particular device.* For example: 1h 29m. For devices with status DOWN, this is the period of their downtime.
{DISCOVERY.RULE.NAME}	→ Discovery notifications	*Name of the discovery rule that discovered the presence or absence of the device or service.*

Macro	Supported in	Description
{DISCOVERY.SERVICE.NAME}	→ Discovery notifications	*Name of the service that was discovered.* For example: HTTP.
{DISCOVERY.SERVICE.PORT}	→ Discovery notifications	*Port of the service that was discovered.* For example: 80.
{DISCOVERY.SERVICE.STATUS}	→ Discovery notifications	*Status of the discovered service:* can be either UP or DOWN.
{DISCOVERY.SERVICE.UPTIME}	→ Discovery notifications	*Time since the last change of discovery status for a particular service.* For example: 1h 29m. For services with status DOWN, this is the period of their downtime.
{ESC.HISTORY}	→ Trigger-based notifications and commands → Internal notifications	*Escalation history. Log of previously sent messages.* Shows previously sent notifications, on which escalation step they were sent and their status (*sent*, *in progress* or *failed*).
{EVENT.ACK.HISTORY}	→ Trigger-based notifications and commands	*Log of acknowledgements on the problem.*
{EVENT.ACK.STATUS}	→ Trigger-based notifications and commands	*Acknowledgement status of the event (Yes/No).*
{EVENT.AGE}	→ Notifications and commands → Internal notifications	*Age of the event that triggered an action.* Useful in escalated messages.
{EVENT.DATE}	→ Notifications and commands → Internal notifications	*Date of the event that triggered an action.*
{EVENT.ID}	→ Notifications and commands → Internal notifications	*Numeric ID of the event that triggered an action.*
{EVENT.RECOVERY.DATE}	→ Trigger-based notifications and commands → Internal notifications	*Date of the recovery event.* Can be used in recovery messages only. Supported since 2.2.0.
{EVENT.RECOVERY.ID}	→ Trigger-based notifications and commands → Internal notifications	*Numeric ID of the recovery event.* Can be used in recovery messages only. Supported since 2.2.0.

Macro	Supported in	Description
{EVENT.RECOVERY.STATUS}	→ Trigger-based notifications and commands → Internal notifications	*Verbal value of the recovery event.* Can be used in recovery messages only. Supported since 2.2.0.
{EVENT.RECOVERY.TIME}	→ Trigger-based notifications and commands → Internal notifications	*Time of the recovery event.* Can be used in recovery messages only. Supported since 2.2.0.
{EVENT.RECOVERY.VALUE}	→ Trigger-based notifications and commands → Internal notifications	*Numeric value of the recovery event.* Can be used in recovery messages only. Supported since 2.2.0.
{EVENT.STATUS}	→ Notifications and commands → Internal notifications	*Verbal value of the event that triggered an action.* Supported since 2.2.0.
{EVENT.TIME}	→ Notifications and commands → Internal notifications	*Time of the event that triggered an action.*
{EVENT.VALUE}	→ Notifications and commands → Internal notifications	*Numeric value of the event that triggered an action.* Supported since 2.2.0.
{HOST.CONN<1-9>}	→ Trigger-based notifications and commands → Internal notifications → Global scripts (including confirmation text) → Map labels[1] → Item key parameters[2] → Host interface IP/DNS → Database monitoring additional parameters[5]	*Host IP address or DNS name, depending on host settings*[3]. Supported in trigger names since 2.0.0.

Macro	Supported in	Description
	→ SSH and Telnet scripts[5] → Web monitoring[6] → Low-level discovery rule filter regular expressions[8] → URL field of dynamic URL screen element[8] → Trigger names and descriptions → Trigger URLs[10]	
{HOST.DESCRIPTION<1-9>}	→ Trigger-based notifications and commands → Internal notifications → Map labels[1]	*Host description.* Supported since 2.4.0.
{HOST.DNS<1-9>}	→ Trigger-based notifications and commands → Internal notifications → Global scripts (including confirmation text) → Map labels[1] → Item key parameters[2] → Host interface IP/DNS → Database monitoring additional parameters[5] → SSH and Telnet scripts[5] → Web monitoring[6] → Low-level discovery rule filter regular expressions[8]	*Host DNS name*[3]. Supported in trigger names since 2.0.0.

Macro	Supported in	Description
	→ URL field of dynamic URL screen element[8] → Trigger names and descriptions → Trigger URLs[10]	
{HOST.HOST<1-9>}	→ Trigger-based notifications and commands → Auto registration notifications → Internal notifications → Global scripts (including confirmation text) → Item key parameters → Map labels[1] → Host interface IP/DNS → Database monitoring additional parameters[5] → SSH and Telnet scripts[5] → Web monitoring[6] → Low-level discovery rule filter regular expressions[8] → URL field of dynamic URL screen element[8] → Trigger names and descriptions → Trigger URLs [10]	*Host name.* `{HOSTNAME<1-9>}` is deprecated.
{HOST.ID<1-9>}	→ Map URLs → URL field of dynamic URL screen element[8]	*Host ID.*

Macro	Supported in	Description
	→ Trigger URLs[10]	
{HOST.IP<1-9>}	→ Trigger-based notifications and commands → Auto registration notifications → Internal notifications → Global scripts (including confirmation text) → Map labels[1] → Item key parameters[2] → Host interface IP/DNS → Database monitoring additional parameters[5] → SSH and Telnet scripts[5] → Web monitoring[6] → Low-level discovery rule filter regular expressions[8] → URL field of dynamic URL screen element[8] → Trigger names and descriptions → Trigger URLs[10]	*Host IP address*[3]. Supported since 2.0.0. `{IPADDRESS<1-9>}` is deprecated.
{HOST.METADATA}	→ Auto registration notifications	*Host metadata*. Used only for active agent auto-registration. Supported since 2.2.0.
{HOST.NAME<1-9>}	→ Trigger-based notifications and commands → Auto registration notifications	*Visible host name*. Supported since 2.0.0.

Macro	Supported in	Description
	→ Internal notifications → Global scripts (including confirmation text) → Map labels[1] → Item key parameters → Host interface IP/DNS → Database monitoring additional parameters[5] → SSH and Telnet scripts[5] → Web monitoring[6] → Low-level discovery rule filter regular expressions[8] → URL field of dynamic URL screen element[8] → Trigger names and descriptions → Trigger URLs[10]	
{HOST.PORT<1-9>}	→ Trigger-based notifications and commands → Auto registration notifications → Internal notifications → Trigger names and descriptions → Trigger URLs[10]	*Host (agent) port*[3]. Supported in auto-registration since 2.0.0. Supported in trigger names, trigger descriptions, internal and trigger-based notifications since 2.2.2.
{HOSTGROUP.ID}	→ Map URLs	*Host group ID.*
{INVENTORY.ALIAS<1-9>}	→ Trigger-based notifications → Internal	*Alias field in host inventory.*

Macro	Supported in	Description
	notifications	
{INVENTORY.ASSET.TAG<1-9>}	→ Trigger-based notifications → Internal notifications	*Asset tag field in host inventory.*
{INVENTORY.CHASSIS<1-9>}	→ Trigger-based notifications → Internal notifications	*Chassis field in host inventory.*
{INVENTORY.CONTACT<1-9>}	→ Trigger-based notifications → Internal notifications	*Contact field in host inventory.* `{PROFILE.CONTACT<1-9>}` is deprecated.
{INVENTORY.CONTRACT.NUMBER<1-9>}	→ Trigger-based notifications → Internal notifications	*Contract number field in host inventory.*
{INVENTORY.DEPLOYMENT.STATUS<1-9>}	→ Trigger-based notifications → Internal notifications	*Deployment status field in host inventory.*
{INVENTORY.HARDWARE<1-9>}	→ Trigger-based notifications → Internal notifications	*Hardware field in host inventory.* `{PROFILE.HARDWARE<1-9>}` is deprecated.
{INVENTORY.HARDWARE.FULL<1-9>}	→ Trigger-based notifications → Internal notifications	*Hardware (Full details) field in host inventory.*
{INVENTORY.HOST.NETMASK<1-9>}	→ Trigger-based notifications → Internal notifications	*Host subnet mask field in host inventory.*
{INVENTORY.HOST.NETWORKS<1-9>}	→ Trigger-based notifications → Internal notifications	*Host networks field in host inventory.*
{INVENTORY.HOST.ROUTER<1-9>}	→ Trigger-based notifications	*Host router field in host inventory.*

Macro	Supported in	Description
	→ Internal notifications	
{INVENTORY.HW.ARCH<1-9>}	→ Trigger-based notifications → Internal notifications	*Hardware architecture field in host inventory.*
{INVENTORY.HW.DATE.DECOMM<1-9>}	→ Trigger-based notifications → Internal notifications	*Date hardware decommissioned field in host inventory.*
{INVENTORY.HW.DATE.EXPIRY<1-9>}	→ Trigger-based notifications → Internal notifications	*Date hardware maintenance expires field in host inventory.*
{INVENTORY.HW.DATE.INSTALL<1-9>}	→ Trigger-based notifications → Internal notifications	*Date hardware installed field in host inventory.*
{INVENTORY.HW.DATE.PURCHASE<1-9>}	→ Trigger-based notifications → Internal notifications	*Date hardware purchased field in host inventory.*
{INVENTORY.INSTALLER.NAME<1-9>}	→ Trigger-based notifications → Internal notifications	*Installer name field in host inventory.*
{INVENTORY.LOCATION<1-9>}	→ Trigger-based notifications → Internal notifications	*Location field in host inventory.* `{PROFILE.LOCATION<1-9>}` is deprecated.
{INVENTORY.LOCATION.LAT<1-9>}	→ Trigger-based notifications → Internal notifications	*Location latitude field in host inventory.*
{INVENTORY.LOCATION.LON<1-9>}	→ Trigger-based notifications → Internal notifications	*Location longitude field in host inventory.*
{INVENTORY.MACADDRESS.A<1-9>}	→ Trigger-based	*MAC address A field in host inventory.* `{PROFILE.MACADDRESS<1-9>}` is

Macro	Supported in	Description
	notifications → Internal notifications	deprecated.
{INVENTORY.MACADDRESS.B<1-9>}	→ Trigger-based notifications → Internal notifications	*MAC address B field in host inventory.*
{INVENTORY.MODEL<1-9>}	→ Trigger-based notifications → Internal notifications	*Model field in host inventory.*
{INVENTORY.NAME<1-9>}	→ Trigger-based notifications → Internal notifications	*Name field in host inventory.* `{PROFILE.NAME<1-9>}` is deprecated.
{INVENTORY.NOTES<1-9>}	→ Trigger-based notifications → Internal notifications	*Notes field in host inventory.* `{PROFILE.NOTES<1-9>}` is deprecated.
{INVENTORY.OOB.IP<1-9>	→ Trigger-based notifications → Internal notifications	*OOB IP address field in host inventory.*
{INVENTORY.OOB.NETMASK<1-9>}	→ Trigger-based notifications → Internal notifications	*OOB subnet mask field in host inventory.*
{INVENTORY.OOB.ROUTER<1-9>}	→ Trigger-based notifications → Internal notifications	*OOB router field in host inventory.*
{INVENTORY.OS<1-9>}	→ Trigger-based notifications → Internal notifications	*OS field in host inventory.* `{PROFILE.OS<1-9>}` is deprecated.
{INVENTORY.OS.FULL<1-9>}	→ Trigger-based notifications → Internal notifications	*OS (Full details) field in host inventory.*

Macro	Supported in	Description
{INVENTORY.OS.SHORT<1-9>}	→ Trigger-based notifications → Internal notifications	*OS (Short) field in host inventory.*
{INVENTORY.POC.PRIMARY.CELL<1-9>}	→ Trigger-based notifications → Internal notifications	*Primary POC cell field in host inventory.*
{INVENTORY.POC.PRIMARY.EMAIL<1-9>}	→ Trigger-based notifications → Internal notifications	*Primary POC email field in host inventory.*
{INVENTORY.POC.PRIMARY.NAME<1-9>}	→ Trigger-based notifications → Internal notifications	*Primary POC name field in host inventory.*
{INVENTORY.POC.PRIMARY.NOTES<1-9>}	→ Trigger-based notifications → Internal notifications	*Primary POC notes field in host inventory.*
{INVENTORY.POC.PRIMARY.PHONE.A<1-9>}	→ Trigger-based notifications → Internal notifications	*Primary POC phone A field in host inventory.*
{INVENTORY.POC.PRIMARY.PHONE.B<1-9>}	→ Trigger-based notifications → Internal notifications	*Primary POC phone B field in host inventory.*
{INVENTORY.POC.PRIMARY.SCREEN<1-9>}	→ Trigger-based notifications → Internal notifications	*Primary POC screen name field in host inventory.*
{INVENTORY.POC.SECONDARY.CELL<1-9>}	→ Trigger-based notifications → Internal notifications	*Secondary POC cell field in host inventory.*
{INVENTORY.POC.SECONDARY.EMAIL<1-9>}	→ Trigger-based notifications → Internal notifications	*Secondary POC email field in host inventory.*

Macro	Supported in	Description
{INVENTORY.POC.SECONDARY.NAME<1-9>}	→ Trigger-based notifications → Internal notifications	*Secondary POC name field in host inventory.*
{INVENTORY.POC.SECONDARY.NOTES<1-9>}	→ Trigger-based notifications → Internal notifications	*Secondary POC notes field in host inventory.*
{INVENTORY.POC.SECONDARY.PHONE.A<1-9>}	→ Trigger-based notifications → Internal notifications	*Secondary POC phone A field in host inventory.*
{INVENTORY.POC.SECONDARY.PHONE.B<1-9>}	→ Trigger-based notifications → Internal notifications	*Secondary POC phone B field in host inventory.*
{INVENTORY.POC.SECONDARY.SCREEN<1-9>}	→ Trigger-based notifications → Internal notifications	*Secondary POC screen name field in host inventory.*
{INVENTORY.SERIALNO.A<1-9>}	→ Trigger-based notifications → Internal notifications	*Serial number A field in host inventory.* `{PROFILE.SERIALNO<1-9>}` is deprecated.
{INVENTORY.SERIALNO.B<1-9>}	→ Trigger-based notifications → Internal notifications	*Serial number B field in host inventory.*
{INVENTORY.SITE.ADDRESS.A<1-9>}	→ Trigger-based notifications → Internal notifications	*Site address A field in host inventory.*
{INVENTORY.SITE.ADDRESS.B<1-9>}	→ Trigger-based notifications → Internal notifications	*Site address B field in host inventory.*
{INVENTORY.SITE.ADDRESS.C<1-9>}	→ Trigger-based notifications → Internal notifications	*Site address C field in host inventory.*

Macro	Supported in	Description
{INVENTORY.SITE.CITY<1-9>}	→ Trigger-based notifications → Internal notifications	*Site city field in host inventory.*
{INVENTORY.SITE.COUNTRY<1-9>}	→ Trigger-based notifications → Internal notifications	*Site country field in host inventory.*
{INVENTORY.SITE.NOTES<1-9>}	→ Trigger-based notifications → Internal notifications	*Site notes field in host inventory.*
{INVENTORY.SITE.RACK<1-9>}	→ Trigger-based notifications → Internal notifications	*Site rack location field in host inventory.*
{INVENTORY.SITE.STATE<1-9>}	→ Trigger-based notifications → Internal notifications	*Site state/province field in host inventory.*
{INVENTORY.SITE.ZIP<1-9>}	→ Trigger-based notifications → Internal notifications	*Site ZIP/postal field in host inventory.*
{INVENTORY.SOFTWARE<1-9>}	→ Trigger-based notifications → Internal notifications	*Software field in host inventory.* `{PROFILE.SOFTWARE<1-9>}` is deprecated.
{INVENTORY.SOFTWARE.APP.A<1-9>}	→ Trigger-based notifications → Internal notifications	*Software application A field in host inventory.*
{INVENTORY.SOFTWARE.APP.B<1-9>}	→ Trigger-based notifications → Internal notifications	*Software application B field in host inventory.*
{INVENTORY.SOFTWARE.APP.C<1-9>}	→ Trigger-based notifications → Internal notifications	*Software application C field in host inventory.*

Macro	Supported in	Description
{INVENTORY.SOFTWARE.APP.D<1-9>}	→ Trigger-based notifications → Internal notifications	*Software application D field in host inventory.*
{INVENTORY.SOFTWARE.APP.E<1-9>}	→ Trigger-based notifications → Internal notifications	*Software application E field in host inventory.*
{INVENTORY.SOFTWARE.FULL<1-9>}	→ Trigger-based notifications → Internal notifications	*Software (Full details) field in host inventory.*
{INVENTORY.TAG<1-9>}	→ Trigger-based notifications → Internal notifications	*Tag field in host inventory.* `{PROFILE.TAG<1-9>}` is deprecated.
{INVENTORY.TYPE<1-9>}	→ Trigger-based notifications → Internal notifications	*Type field in host inventory.* `{PROFILE.DEVICETYPE<1-9>}` is deprecated.
{INVENTORY.TYPE.FULL<1-9>}	→ Trigger-based notifications → Internal notifications	*Type (Full details) field in host inventory.*
{INVENTORY.URL.A<1-9>}	→ Trigger-based notifications → Internal notifications	*URL A field in host inventory.*
{INVENTORY.URL.B<1-9>}	→ Trigger-based notifications → Internal notifications	*URL B field in host inventory.*
{INVENTORY.URL.C<1-9>}	→ Trigger-based notifications → Internal notifications	*URL C field in host inventory.*
{INVENTORY.VENDOR<1-9>}	→ Trigger-based notifications → Internal notifications	*Vendor field in host inventory.*

Macro	Supported in	Description
{ITEM.DESCRIPTION<1-9>}	→ Trigger-based notifications → Internal notifications	*Description of the Nth item in the trigger expression that caused a notification.* Supported since 2.0.0.
{ITEM.ID<1-9>}	→ Trigger-based notifications → Internal notifications	*Numeric ID of the Nth item in the trigger expression that caused a notification.* Supported since 1.8.12.
{ITEM.KEY<1-9>}	→ Trigger-based notifications → Internal notifications	*Key of the Nth item in the trigger expression that caused a notification.* Supported since 2.0.0. `{TRIGGER.KEY}` is deprecated.
{ITEM.KEY.ORIG<1-9>}	→ Trigger-based notifications → Internal notifications	*Original key (with macros not expanded) of the Nth item in the trigger expression that caused a notification.* Supported since 2.0.6.
{ITEM.LASTVALUE<1-9>}	→ Trigger-based notifications → Trigger names and descriptions	*The latest value of the Nth item in the trigger expression that caused a notification.* It will resolve to *UNKNOWN* in the frontend if the latest history value has been collected more than the *ZBX_HISTORY_PERIOD* time ago (defined in defines.inc.php). Supported since 1.4.3. It is alias to `{{HOST.HOST}:{ITEM.KEY}.last()}`
{ITEM.LOG.AGE<1-9>}	→ Trigger-based notifications	*Age of the log item event.*
{ITEM.LOG.DATE<1-9>}	→ Trigger-based notifications	*Date of the log item event.*
{ITEM.LOG.EVENTID<1-9>}	→ Trigger-based notifications	*ID of the event in the event log.* For Windows event log monitoring only.
{ITEM.LOG.NSEVERITY<1-9>}	→ Trigger-based notifications	*Numeric severity of the event in the event log.* For Windows event log monitoring only.
{ITEM.LOG.SEVERITY<1-9>}	→ Trigger-based notifications	*Verbal severity of the event in the event log.* For Windows event log monitoring only.
{ITEM.LOG.SOURCE<1-9>}	→ Trigger-based notifications	*Source of the event in the event log.* For Windows event log monitoring only.

Macro	Supported in	Description
{ITEM.LOG.TIME<1-9>}	→ Trigger-based notifications	*Time of the log item event.*
{ITEM.NAME<1-9>}	→ Trigger-based notifications → Internal notifications	*Name of the Nth item in the trigger expression that caused a notification.*
{ITEM.NAME.ORIG<1-9>}	→ Trigger-based notifications → Internal notifications	*Original name (with macros not expanded) of the Nth item in the trigger expression that caused a notification.* Supported since 2.0.6.
{ITEM.STATE<1-9>}	→ Item-based internal notifications	*The latest state of the Nth item in the trigger expression that caused a notification.* Possible values: **Not supported** and **Normal**. Supported since 2.2.0.
{ITEM.VALUE<1-9>}	→ Trigger-based notifications → Trigger names and descriptions	Resolved to either: 1) the historical (at-the-time-of-event) value of the Nth item in the trigger expression, if used in the context of trigger status change, for example, when displaying events or sending notifications. 2) the latest value of the Nth item in the trigger expression, if used without the context of trigger status change, for example, when displaying a list of triggers in a pop-up selection window. In this case works the same as {ITEM.LASTVALUE} In the first case it will resolve to *UNKNOWN* if the history value has already been deleted or has never been stored. In the second case, and in the frontend only, it will resolve to *UNKNOWN* if the latest history value has been collected more than the *ZBX_HISTORY_PERIOD* time ago (defined in defines.inc.php). Supported since 1.4.3.
{LLDRULE.DESCRIPTION}	→ LLD-rule based internal notifications	*Description of the low-level discovery rule which caused a notification.* Supported since 2.2.0.
{LLDRULE.ID}	→ LLD-rule based internal notifications	*Numeric ID of the low-level discovery rule which caused a notification.* Supported since 2.2.0.
{LLDRULE.KEY}	→ LLD-rule based internal notifications	*Key of the low-level discovery rule which caused a notification.* Supported since 2.2.0.

Macro	Supported in	Description
{LLDRULE.KEY.ORIG}	→ LLD-rule based internal notifications	*Original key (with macros not expanded) of the low-level discovery rule which caused a notification.* Supported since 2.2.0.
{LLDRULE.NAME}	→ LLD-rule based internal notifications	*Name of the low-level discovery rule which caused a notification.* Supported since 2.2.0.
{LLDRULE.NAME.ORIG}	→ LLD-rule based internal notifications	*Original name (with macros not expanded) of the low-level discovery rule which caused a notification.* Supported since 2.2.0.
{LLDRULE.STATE}	→ LLD-rule based internal notifications	*The latest state of the low-level discovery rule.* Possible values: **Not supported** and **Normal**. Supported since 2.2.0.
{MAP.ID}	→ Map URLs	*Network map ID.*
{PROXY.DESCRIPTION<1-9>}	→ Notifications and commands → Internal notifications	*Proxy description of the Nth item in the trigger expression that caused a notification.* Supported since 2.4.0.
{PROXY.NAME<1-9>}	→ Notifications and commands → Internal notifications	*Proxy name of the Nth item in the trigger expression that caused a notification.* Supported since 1.8.4.
{TIME}	→ Notifications and commands → Internal notifications	*Current time in hh:mm:ss.*
{TRIGGER.DESCRIPTION}	→ Trigger-based notifications → Trigger-based internal notifications	*Trigger description.* Supported since 2.0.4. Starting with 2.2.0, all macros supported in a trigger description will be expanded if `{TRIGGER.DESCRIPTION}` is used in notification text. `{TRIGGER.COMMENT}` is deprecated.
{TRIGGER.EVENTS.ACK}	→ Trigger-based notifications → Map labels[1].	*Number of acknowledged events for a map element in maps, or for the trigger which generated current event in notifications.* Supported since 1.8.3.
{TRIGGER.EVENTS.PROBLEM.ACK}	→ Trigger-based notifications → Map labels[1].	*Number of acknowledged PROBLEM events for all triggers disregarding their state.* Supported since 1.8.3.
{TRIGGER.EVENTS.PROBLEM.UNACK}	→ Trigger-based notifications → Map labels[1].	*Number of unacknowledged PROBLEM events for all triggers disregarding their state.* Supported since 1.8.3.

Macro	Supported in	Description
{TRIGGER.EVENTS.UNACK}	→ Trigger-based notifications → Map labels[1]	*Number of unacknowledged events for a map element in maps, or for the trigger which generated current event in notifications.* Supported in map element labels since 1.8.3.
{TRIGGER.HOSTGROUP.NAME}	→ Trigger-based notifications → Trigger-based internal notifications → Map labels[1]	*A sorted (by SQL query), comma-space separated list of host groups in which the trigger is defined.* Supported since 2.0.6.
{TRIGGER.PROBLEM.EVENTS.PROBLEM.ACK}	→ Map labels[1]	*Number of acknowledged PROBLEM events for triggers in PROBLEM state.* Supported since 1.8.3.
{TRIGGER.PROBLEM.EVENTS.PROBLEM.UNACK}	→ Map labels[1]	*Number of unacknowledged PROBLEM events for triggers in PROBLEM state.* Supported since 1.8.3.
{TRIGGER.EXPRESSION}	→ Trigger-based notifications → Trigger-based internal notifications	*Trigger expression.* Supported since 1.8.12.
{TRIGGER.ID}	→ Trigger-based notifications → Trigger-based internal notifications → Map URLs → Trigger URLs	*Numeric trigger ID which triggered this action.* Supported in trigger URLs since Zabbix 1.8.8.
{TRIGGER.NAME}	→ Trigger-based notifications → Trigger-based internal notifications	*Name of the trigger.*
{TRIGGER.NAME.ORIG}	→ Trigger-based notifications → Trigger-based internal notifications	*Original name (with macros not expanded) of the trigger.* Supported since 2.0.6.

Macro	Supported in	Description
{TRIGGER.NSEVERITY}	→ Trigger-based notifications → Trigger-based internal notifications	*Numerical trigger severity.* Possible values: 0 - Not classified, 1 - Information, 2 - Warning, 3 - Average, 4 - High, 5 - Disaster. Supported starting from Zabbix 1.6.2.
{TRIGGER.SEVERITY}	→ Trigger-based notifications → Trigger-based internal notifications	*Trigger severity name.* Can be defined in *Administration → General → Trigger severities.*
{TRIGGER.STATE}	→ Trigger-based internal notifications	*The latest state of the trigger.* Possible values: **Unknown** and **Normal**. Supported since 2.2.0.
{TRIGGER.STATUS}	→ Trigger-based notifications	*Current trigger value.* Can be either PROBLEM or OK. `{STATUS}` is deprecated.
{TRIGGER.TEMPLATE.NAME}	→ Trigger-based notifications → Trigger-based internal notifications	*A sorted (by SQL query), comma-space separated list of templates in which the trigger is defined, or *UNKNOWN* if the trigger is defined in a host.* Supported since 2.0.6.
{TRIGGER.URL}	→ Trigger-based notifications → Trigger-based internal notifications	*Trigger URL.*
{TRIGGER.VALUE}	→ Trigger-based notifications → Trigger expressions	*Current trigger numeric value*: 0 - trigger is in OK state, 1 - trigger is in PROBLEM state.
{TRIGGERS.UNACK}	→ Map labels[1].	*Number of unacknowledged triggers for a map element, disregarding trigger state.* A trigger is considered to be unacknowledged if at least one of its PROBLEM events is unacknowledged.
{TRIGGERS.PROBLEM.UNACK}	→ Map labels[1].	*Number of unacknowledged PROBLEM triggers for a map element.* A trigger is considered to be unacknowledged if at least one of its PROBLEM events is unacknowledged.

Macro	Supported in	Description
		Supported since 1.8.3.
{TRIGGERS.ACK}	→ Map labels[1]	*Number of acknowledged triggers for a map element, disregarding trigger state.* A trigger is considered to be acknowledged if all of it's PROBLEM events are acknowledged. Supported since 1.8.3.
{TRIGGERS.PROBLEM.ACK}	→ Map labels[1]	*Number of acknowledged PROBLEM triggers for a map element.* A trigger is considered to be acknowledged if all of it's PROBLEM events are acknowledged. Supported since 1.8.3.
{host:key.func(param)}	→ Trigger-based notifications → Map labels[1] [4] → Graph names[7] → Trigger expressions[9]	*Simple macros, as used in building trigger expressions.*
{$MACRO}	→ Trigger-based notifications[8] → Trigger-based internal notifications[8] → Global scripts (including confirmation text) → Item key parameters → Item names → Host interface IP/DNS → Database monitoring additional parameters[5] → SSH and Telnet scripts[5] → Web monitoring[6] → Low-level discovery rule filter regular expressions[8]	*User-definable macros.* Supported in item and trigger names since 1.8.4. Supported in global script commands and confirmation texts since Zabbix 2.2.0.

Macro	Supported in	Description
	→ URL field of dynamic URL screen element[8] → Trigger expressions (only in constants and function parameters) → Trigger names and descriptions → Trigger URLs[10] → See also: Additional support for user macros	

Footnotes

[1] Macros for map labels are supported since 1.8.

[2] Since Zabbix 2.0.3, the HOST.* macros supported in item key parameters will only work in item types that have interfaces, i.e. they will not work for types "Zabbix agent (active)", "Calculated" etc.

[3] Depending on the context, the macro will be expanded differently. In **2.0.1** or earlier versions, in the context of remote command execution, GUI scripts, item key parameters, and interface IP/DNS fields only the main agent interface will be considered as the source of information. Since **2.0.2** and for web scenarios since **2.2.0** the macro will use the main agent interface, however, if it is not present, the main SNMP interface will be used. If SNMP is also not present, the main JMX interface will be used. If JMX is not present either, the main IPMI interface will be used. In item key **parameters** the interface that is selected for the item will be used, since 2.0.3.

[4] Only the **avg**, **last**, **max** and **min** functions, with seconds as parameter are supported in this macro in map labels.

[5] Supported since 2.0.3.

[6] Supported since Zabbix 2.2.0. `{HOST.*}` macros and user-defined macros `{$MACRO}` are supported in web scenario *Name*, *Variables*, *Headers*, *SSL certificate file* and *SSL key file* fields and in scenario step *Name*, *URL*, *Post*, *Headers* and *Required string* fields. `{$MACRO}` is also supported in web scenario *Agent*, *HTTP proxy*, *Authentication* (user and password) and *SSL key password* fields and in the scenario step *Required status codes* field.

[7] Supported since Zabbix 2.2.0. Only the **avg**, **last**, **max** and **min** functions, with seconds as parameter are supported within this macro in graph names. The {HOST.HOST<1-9>} macro can be used as host within the macro. For example:

- {Cisco switch:ifAlias[{#SNMPINDEX}].last()}
- {{HOST.HOST}:ifAlias[{#SNMPINDEX}].last()}

[8] Supported since 2.4.0.

[9] While supported to build trigger expressions, simple macros may not be used inside each other.

[10] Supported since 3.0.0.

Indexed macros

The indexed macro syntax of {MACRO**<1-9>**} is limited to the context of **trigger expressions**. It can be used to reference hosts in the order in which they appear in the expression. Macros like {HOST.IP1}, {HOST.IP2}, {HOST.IP3} will resolve to the IP of the first, second and third host in the trigger expression (providing the trigger expression contains those hosts).

Additionally the {HOST.HOST<1-9>} macro is also supported within the `{host:key.func(param)}` macro in **graph names**. For example, `{{HOST.HOST2}:key.func()}` in the graph name will refer to the host of the second item in the graph.

Use macros **without** index (i. e.{HOST.HOST}, {HOST.IP}, etc) in all other contexts.

Additional support for user macros

In addition to the locations listed, user-definable macros since Zabbix 2.0 are supported in numerous other locations:

- Hosts
    - Interface IP/DNS
    - Interface port
- Passive proxy
    - Interface port
- Items
    - SNMPv3 context name
    - SNMPv3 security name
    - SNMPv3 auth pass
    - SNMPv3 priv pass
    - SNMPv1/v2 community
    - SNMP OID
    - SNMP port
    - SSH username
    - SSH public key
    - SSH private key
    - SSH password
    - Telnet username
    - Telnet password
    - Calculated item formula
    - Trapper item "Allowed hosts" field (*since Zabbix 2.2*)
- Discovery
    - SNMPv3 context name
    - SNMPv3 security name
    - SNMPv3 auth pass
    - SNMPv3 priv pass
    - SNMPv1/v2 community
    - SNMP OID

Macros used in low-level discovery

There is a type of macro used within the low-level discovery function - **{#MACRO}**. It is a macro that is used in an LLD rule and returns real values of file system names, network interfaces and SNMP OIDs.

These macros can be used for creating item, trigger and graph *prototypes*. Then, when discovering real file systems, network interfaces etc., these macros are substituted with real values and are the basis for creating real items, triggers and graphs.

These macros are also used in creating host and host group prototypes in virtual machine discovery.

LLD macros can be used:

- for item prototypes in
    - names
    - key parameters
    - units
    - SNMP OIDs
    - IPMI sensor fields
    - calculated item formulas
    - SSH and Telnet scripts
    - database monitoring SQL queries
    - descriptions (supported since 2.2.0)
- for trigger prototypes in
    - names
    - expressions (insofar as when referencing an item key prototype and as standalone constants)
    - URLs (supported since 3.0.0)
    - descriptions (supported since 2.2.0)
- for graph prototypes in
    - names
- for host prototypes (supported since 2.2.0) in
    - names
    - visible names
    - host group prototype names
    - (see the full list)

In all those places LLD macros can be used inside user macro context.

Some low-level discovery macros come "pre-packaged" with the LLD function in Zabbix - {#FSNAME}, {#FSTYPE}, {#IFNAME}, {#SNMPINDEX}, {#SNMPVALUE}. However, adhering to these names is not compulsory when creating a custom low-level discovery rule. Then you may use any other LLD macro name and refer to that name.
