



OPEN NETWORKING  
FOUNDATION

# Output match Extension

*Version 0.1*

---

December 23, 2014



## Disclaimer

THIS SPECIFICATION IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE. Without limitation, ONF disclaims all liability, including liability for infringement of any proprietary rights, relating to use of information in this specification and to the implementation of this specification, and ONF disclaims all liability for cost of procurement of substitute goods or services, lost profits, loss of use, loss of data or any incidental, consequential, direct, indirect, or special damages, whether under contract, tort, warranty or otherwise, arising in any way out of use or reliance upon this specification or any information herein.

No license, express or implied, by estoppel or otherwise, to any Open Networking Foundation or Open Networking Foundation member intellectual property rights is granted herein.

Except that a license is hereby granted by ONF to copy and reproduce this specification for internal use only.

Contact the Open Networking Foundation at <http://www.opennetworking.org> for information on specification licensing through membership agreements.

Any marks and brands contained herein are the property of their respective owners.

**WITHOUT LIMITING THE DISCLAIMER ABOVE, THIS SPECIFICATION OF THE OPEN NETWORKING FOUNDATION ("ONF") IS SUBJECT TO THE ROYALTY FREE, REASONABLE AND NONDISCRIMINATORY ("RANDZ") LICENSING COMMITMENTS OF THE MEMBERS OF ONF PURSUANT TO THE ONF INTELLECTUAL PROPERTY RIGHTS POLICY. ONF DOES NOT WARRANT THAT ALL NECESSARY CLAIMS OF PATENT WHICH MAY BE IMPLICATED BY THE IMPLEMENTATION OF THIS SPECIFICATION ARE OWNED OR LICENSABLE BY ONF'S MEMBERS AND THEREFORE SUBJECT TO THE RANDZ COMMITMENT OF THE MEMBERS.**

## Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>How it works</b>	<b>3</b>
<b>3</b>	<b>Output match Experimenter ID</b>	<b>3</b>
<b>4</b>	<b>Output match field</b>	<b>3</b>

## 1 Introduction

This document describes an ONF extension for OpenFlow version 1.3.X that match the output port from the action set.

## 2 How it works

A new OXM match field is defined. This field map to the output port decision of the action set, and enable output port based processing.

The Action Set Output match field `ONFOXM_ET_ACTSET_OUTPUT` can be used to match the forwarding actions embedded in the action set of the packet, and in most cases can be used to match packet based on their output port. If the action set contains an output action and no group action, this field equals the output port from that action. Else, the field equals its initial value, `OFPP_TABLE`.

The match field `ONFOXM_ET_ACTSET_OUTPUT` is a pipeline field, however it should not be included by the switch in the match field of a packet-in message.

## 3 Output match Experimenter ID

The Experimenter ID of this extension is:

```
ONF_EXPERIMENTER_ID = 0x4F4E4600
```

## 4 Output match field

This extension defines the following experimenter oxm types:

```
/* OXM types */
enum onf_oxm_exp_type {
    ONFOXM_ET_ACTSET_OUTPUT = 43, /* Output port from action set Metadata. */
};
```

The OXM fields have the size, prerequisites and masking capability specified in the following table.

Field	Bits	Mask	Pre-requisite	Description
ONFOXM_ET_ACTSET_OUTPUT	32	No	None	Output port from action set Metadata.

Table 1: Output port match fields details.

The ONFOXM\_ET\_ACTSET\_OUTPUT oxm type uses the following action structure :

```

/* Structure for OXM field output match. */
struct onf_oxm_actset_output {
    uint32_t    oxm_header;    /* oxm_class = OFPXM_C_EXPERIMENTER,
                               oxm_field = ONFOXM_ET_ACTSET_OUTPUT. */
    uint32_t    experimenter; /* ONF_EXPERIMENTER_ID. */
    uint64_t    output;       /* Output port from action set Metadata. */
};
OFP_ASSERT(sizeof(struct onf_oxm_actset_output) == 16);

```

The `oxm_header` field must be set with class `OFPXM_C_EXPERIMENTER`, field type `ONFOXM_ET_ACTSET_OUTPUT` and the proper OXM length and mask bit.

The `experimenter` field is the Experimenter ID (see 3).

The `output` field is the output port from the action set to match.